PROYECTO FIN DE CARRERA

INGENIERÍA DE TELECOMUNICACIÓN

AVeMaCS:

Desarrollo de una Aplicación para la Gestión de la Verificación de Sistemas Críticos

Autor: JULIO ESCRIBANO BARRENO

Tutor: MARISOL GARCÍA VALLS

Leganés, Marzo de 2013

"Página dejada en blanco intencionadamente"

Título: AVeMaCS: Desarrollo de una Herramienta para la Gestión de la Verificación en Sistemas Críticos

Autor: Julio Escribano Barreno

Director: Marisol García Valls

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día __ de _____ de 20__ en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO PRESIDENTE

"Página dejada en blanco intencionadamente"

AGRADECIMIENTOS Y DEDICATORIA

A Marisol, mi tutora, por su apoyo y los ánimos que me ha dado a lo largo de este proyecto.

A Indra, por proporcionarme gran parte de la experiencia que me ha llevado a realizar este trabajo.

A todos mis compañeros y amigos, que me han acompañado estos años durante el camino.

A mis padres, por el esfuerzo que han hecho para hacer posible que llegara hasta aquí, sufriendo además muchos enfados, flustraciones y lamentos que me acompañaron casi todo ese tiempo.

A mi hermana, por servirme de guía sin saberlo.

A Neli por su paciencia, por su comprensión, y por no haber podido dedicarle todo el tiempo que se merece durante la realización de este trabajo.

A Gonzalo, por ser la alegría de cada día.

Al futuro.

"Página dejada en blanco intencionadamente"

ÍNDICE GENERAL

Capítulo Descripción Página

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

"Página dejada en blanco intencionadamente"

CAPÍTULO 1: CONTEXTO Y MOTIVACIÓN DEL PROYECTO

En este capítulo se describirá el contexto en el cual se enmarca este trabajo y la motivación que ha llevado al desarrollo del mismo.

1.1. CONTEXTO

La importancia del software ha ido aumentando según ha cambiado la filosofía de los computadores en entornos de aplicaciones aeronáuticas. Actualmente, las aeronaves se basan cada vez más en los computadores y menos en la mecánica, con lo que la cantidad de software que se encuentra en sus sistemas ha ido aumentando con los años.

Un mal funcionamiento en los equipos embarcados puede impedir realizar una función correctamente, con distintas consecuencias. Podría afectar a la funcionalidad, como por ejemplo un fallo en la radio, pero también podría afectar a la operación segura de la aeronave, como por ejemplo si se produjese una pérdida total en las comunicaciones y navegación.

Por tanto, un fallo del software podría hacer que un equipo o sistema funcione de manera incorrecta, lo que podría afectar a la seguridad del vuelo.

Esto ha conllevado que los sistemas aeronáuticos deban afrontar unos requisitos de seguridad y fiabilidad extremadamente exigentes debido a las circunstancias especiales que los rodean.

La seguridad del software ha ido cobrando relevancia a lo largo de los últimos años debido a la incorporación de computadores en los sistemas críticos. Por ello, entre otras cosas, es imprescindible seguir un proceso de desarrollo exhaustivo y de calidad que comprenda además las actividades necesarias para incrementar la seguridad y la fiabilidad relacionada con el software.

El uso de estas técnicas no está suficientemente extendido, lo que se debe, en parte, a que se trata de técnicas costosas, tanto por su grado de dificultad como por motivos económicos.

La IEC 61508 (*Functional safety of electrical/electronic/programmable electronic safety-related systems*, Ref. [1]) define la seguridad como “la ausencia de niveles de riesgo inaceptables”. En este sentido, la seguridad está relacionada estrechamente con la fiabilidad.

La filosofía de la seguridad en los sistemas críticos se puede definir en tres puntos:

- El riesgo cero es imposible de conseguir.
- La seguridad (*safety*) debe ser considerada desde el inicio
- Los riesgos no tolerables deben ser reducidos.

En la Figura 1 se puede ver las causas de fallos en los sistemas, según un estudio de la Comisión de Salud y Seguridad de Gran Bretaña (Ref. [2]).

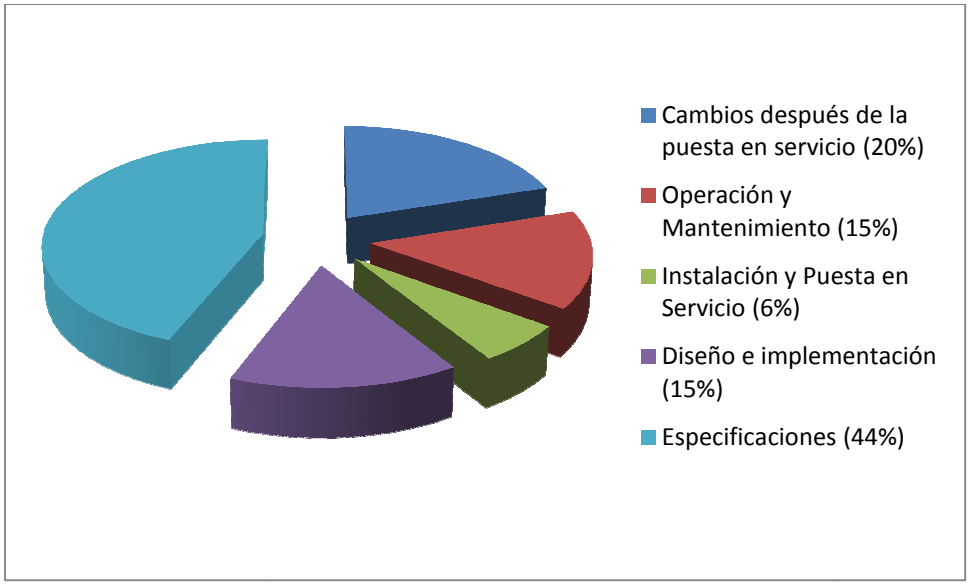


Figura 1. Causa principal de los fallos en sistemas por fase del ciclo de vida

Según estos datos, el 60% de los fallos podría evitarse durante el diseño y desarrollo del sistema.

El 44% de los fallos corresponde con errores en las especificaciones, lo que coincide con lo que decía F. Brooks en su artículo "No Silver Bullet — Essence and Accidents of Software Engineering" (Ref. [3]). En ese artículo hablaba de que "la parte más dura de hacer software es, precisamente, decidir lo que hacer. Ninguna otra de las partes en que se divide este trabajo es tan difícil como establecer los requisitos técnicos detallados, incluyendo todas las interfaces con las personas, las máquinas y los otros sistemas de software. Ninguna otra parte del trabajo destroza tanto el resultado final si se hace mal. Ninguna otra parte resulta tan difícil de rectificar a posteriori."

Existen una serie de normativas (algunas de ellas descritas en el Capítulo 0) que proporcionan los instrumentos necesarios para alcanzar un nivel de seguridad aceptable, definiendo los procesos y evidencias necesarios, desde la planificación, pasando por la definición de requisitos, hasta la codificación.

1.1. OBJETIVOS Y MOTIVACIÓN DEL PROYECTO

Como es de esperar, el incremento de los costes y de los tiempos de desarrollo depende directamente de la contribución que tiene el software a las condiciones de fallo del sistema. Esta contribución se traduce en un nivel de criticidad que está definido de acuerdo a la normativa aplicable.

En la Figura 2 se puede ver la relación teórica entre los costes y el tiempo de desarrollo según los niveles de criticidad definidos en la DO-178B (*Software Considerations in Airborne Systems and Equipement Certification*, Ref. [4]) (Ver capítulo 0 para más detalles).

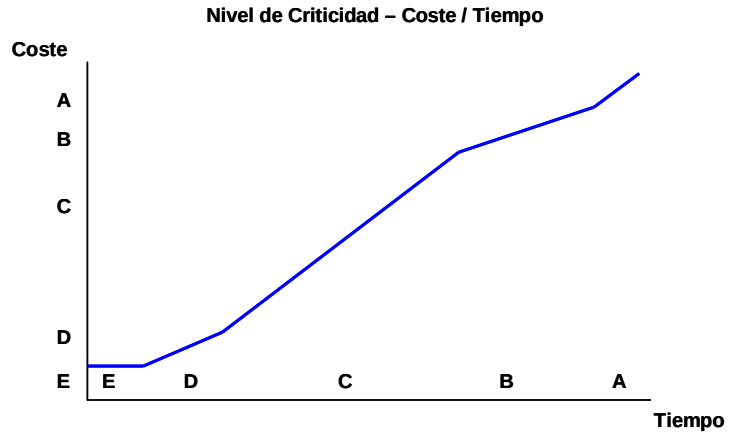


Figura 2. Coste/Tiempo según el nivel de criticidad del software

En la Figura 3 se puede ver el incremento de los costes según el nivel de criticidad.

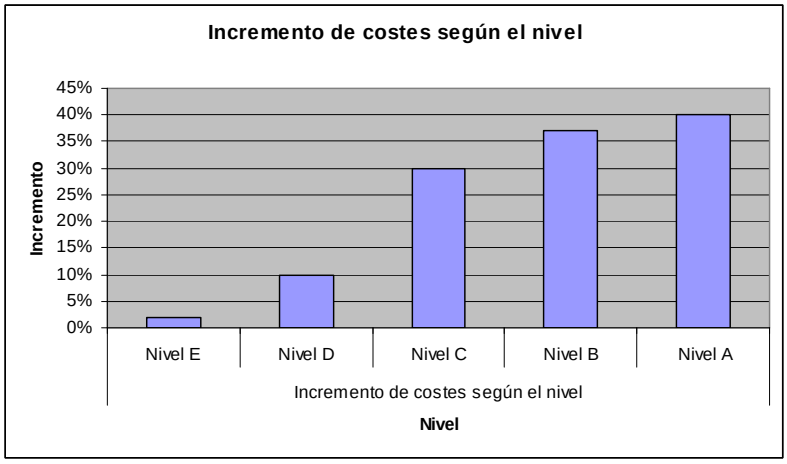


Figura 3. Incremento de costes según el nivel de criticidad

Aunque el aumento de los costes en el desarrollo de los proyectos sólo debería ser habitualmente de entre un 25% y un 40%, la media de incremento de costes es de entre el 75% y el 150%, debido a varios factores, como pueden ser la formación, la mala planificación o la aplicación de unos procesos no óptimos para el desarrollo de sistemas críticos.

En el ámbito de este tipo de proyectos, existe un proceso continuo de investigación para la reducción de costes de la realización de estas tareas, que muchas veces pasan por definir metodologías y automatizar procesos.

La definición inequívoca de las distintas etapas dentro de los procesos del ciclo de vida conlleva que los distintos equipos tengan claro las actividades relacionadas con el cumplimiento de la normativa aplicable. Esto hace que se detecten las no conformidades lo antes posible, y se pueda modificar lo necesario para el cumplimiento.

Una forma de contribuir a estos objetivos es la utilización de herramientas que optimicen el seguimiento de los procesos. Por ejemplo, si una herramienta define el ciclo de vida, con entradas y salidas para cada etapa, se facilitará el aprendizaje, así como la aplicación y el seguimiento del cumplimiento con la normativa.

Esta situación implica que sea de gran utilidad una investigación para demostrar las ventajas reales de la optimización de los procesos a través de herramientas de soporte al desarrollo y la verificación.

Dentro de este proceso de investigación se ha desarrollado la herramienta objeto de este proyecto.

Como base de cumplimiento de estos objetivos, se proponen las siguientes metas:

- Cubrir las actividades de gestión de la verificación
- Desarrollo de un entorno colaborativo
- Facilitar la integración con el resto de procesos del ciclo de vida
- Posibilidad de ampliaciones futuras

a. Cubrir las actividades de gestión de la verificación

Uno de los principales objetivos es cubrir las actividades de gestión de la verificación, de tal forma que sea posible gestionar las tareas y almacenar los resultados obtenidos. La información del proceso de verificación estará almacenada y servirá como evidencia de que las distintas actividades del proceso de verificación se están llevando a cabo según la normativa aplicable en cada caso.

b. Desarrollo de un entorno colaborativo

Si se desarrolla un entorno colaborativo, la información estará disponible para todos los participantes en los proyectos, de tal forma que cualquier parte interesada podrá consultar fácilmente el estado de cada una de las tareas que forman el ciclo de vida de desarrollo según la normativa aplicable. Esta disponibilidad aportará claridad en los procesos y facilitará el flujo de información entre los distintos equipos de trabajo.

c. Facilitar la integración con el resto de procesos del ciclo de vida

La tecnología utilizada debe permitir que sea posible integrar de alguna forma la información contenida con otros procesos del ciclo de vida, como son la gestión de la configuración, gestión de incidencias.

d. Posibilidad de ampliaciones futuras

Es necesario que la herramienta sea ampliable para cubrir posibles necesidades futuras o para ampliar el alcance de las actividades para las que se define en este momento.

1.2. LA HERRAMIENTA AVEMACS COMO PARTE DEL PROCESO DE CERTIFICACIÓN

Una de las bases del proceso de desarrollo es el seguimiento de un modelo del ciclo de vida. Los estándares que existen en la industria proponen diferentes modelos que incluyen en cada etapa del ciclo de vida todas las actividades relacionadas con la seguridad que son necesarias para llevar a cabo.

La herramienta “AVeMaCS” (acrónimo de *Asisted VErification MAnagement for Critical Systems*) permite la gestión de la verificación software a lo largo del ciclo de vida de desarrollo del software. Esta herramienta facilita un entorno de trabajo colaborativo y con la información disponible en todo momento. De esta forma se puede conocer el estado del cumplimiento de la verificación en cualquier momento del desarrollo, así como el grado de progreso y las actividades pendientes.

Las normativas que se han utilizado como base de este desarrollo son la DO-178B/ED-12B (Ref. [4]) y la DO-278/ED-109 (*Guidelines for Communication, Navigation Surveillance, and Air Traffic Management (CNS/ATM) Systems Software Integrity Assurance*, Ref. [5]). La posibilidad de personalización de la herramienta permite que se puedan incorporar otras

normativas con facilidad. Dentro de estas normativas que se podrían incorporar, sería posible añadir otras normativas relacionadas con la verificación del hardware (o Firmware), como es el caso de la DO-254 (*Design Assurance Guidance for Airborne Electronic Hardware*, Ref. [6]).

La herramienta permite la definición de varios proyectos a un mismo tiempo para su gestión independiente. También tiene definidos varios tipos de usuarios o roles, de tal forma que dependiendo de esta definición, los distintos usuarios tendrán permitidas unas determinadas acciones u otras.

1.3. ALCANCE DE LA HERRAMIENTA

El proceso de verificación es un proceso integral, que debe estar integrado y coordinado con el resto de procesos del ciclo de vida de desarrollo, tal y como se describirá en el apartado 0.

AVeMaCS tiene como objetivo centralizar los procesos de verificación software en un único entorno, accesible para todos los participantes en un proyecto que tengan que cumplir una normativa relacionada con la verificación.

El entorno funcional en el que se mueve la herramienta se puede ver en la Figura 4.

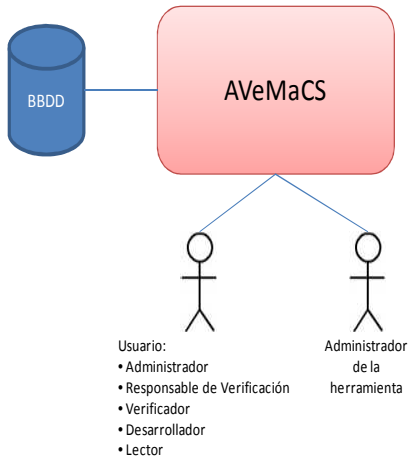


Figura 4. Entorno funcional de la herramienta

La herramienta está desarrollada en HTML, PHP y MySQL, de tal forma que se puede acceder a través de cualquier navegador web. Los usuarios que acceden a la herramienta pueden tener distintos roles, dependiendo de las actividades que realicen dentro de cada proyecto. Siempre existirá al menos un administrador de la herramienta. La herramienta está conectada con una base de datos en MySQL donde se almacenarán todos los datos de la aplicación, como proyectos, documentos, listas de comprobación, comentarios.

Uno de los objetivos a perseguir en la definición y desarrollo de esta herramienta es, por tanto, facilitar la integración con el resto de procesos implicados en la herramienta.

1.4. CONVENCIONES Y ESTRUCTURA

1.4.1. Convenciones

Para entender mejor el término *seguridad* que del que se habla en esta memoria, se debe tener en cuenta cierto matiz entre los términos anglosajones *security* y *safety*. Estas palabras tienen distinto significado en inglés, pero en español, la traducción es la misma: *seguridad*.

Security es la prevención de actos intencionados que afectan a materiales o personas (secuestros, bombas, etc.). *Safety* se relaciona con la prevención de eventos accidentales que pueden afectar a material o personas (diseño del equipo, mantenimiento, etc.). En algunos casos, *safety* puede aparecer traducido como *seguridad física*, pero esta traducción puede llevar a malentendidos, por lo que en este trabajo lo traduciremos como *safety* para evitar malas interpretaciones.

En la Figura 5, se muestra la reproducción literal de un cartel en un aeropuerto inglés, bastante representativa para ver la diferencia entre estos dos conceptos.

La consideración de la seguridad en el desarrollo de sistemas críticos se referirá, por tanto, al *safety*. Se utilizará por tanto este término anglosajón en la memoria del proyecto para enfatizar el significado de la seguridad que se está teniendo en cuenta.

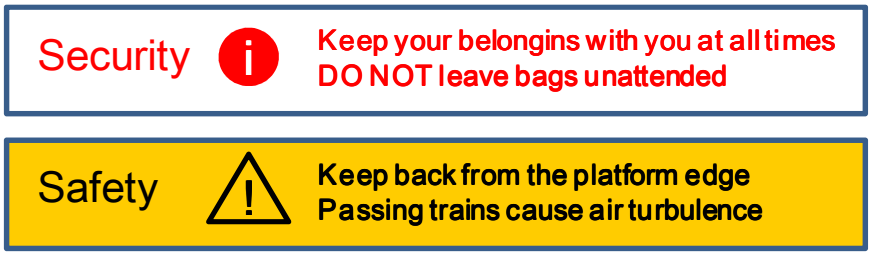


Figura 5. Diferencia entre safety y security

1.4.2. Estructura

La memoria de este proyecto está estructurada en los siguientes capítulos:

- Capítulo 1: Contexto y Motivación del Proyecto: se define el contexto en el que se enmarca el presente trabajo, así como la motivación que ha llevado al desarrollo del mismo. Se definen también los objetivos a conseguir, la estructura y el significado de los acrónimos que se utilizan a lo largo del proyecto.
- Capítulo 2: Normativas y Procesos: En este capítulo se describen las normativas que pretende cubrir el proyecto, así como los procesos implementados en el desarrollo del mismo.
- Capítulo 3: AVEMACS: Herramienta para la gestión de la verificación software en sistemas críticos: Se presentará la especificación general de la herramienta, así como el entorno gráfico y una guía de uso.
- Capítulo 4: Caso práctico: Aquí se presentarán los resultados de utilización de la herramienta en un caso real.
- Capítulo 5: Conclusiones: Se revisarán los objetivos planteados al inicio del proyecto así como la forma en que se han conseguido.

- Capítulo 6: Bibliografía: Presenta la bibliografía que se ha utilizado para el desarrollo de la memoria de este proyecto.
- Capítulo 7: Presupuesto estimado: Incluye un presupuesto estimado para llevar a cabo lo presentado en este trabajo.

1.4.3. Acrónimos

La siguiente lista contiene los acrónimos utilizados en esta memoria de proyecto. La columna "traducción" no siempre se corresponde con una traducción literal del término en inglés, sino que es una adaptación al lenguaje español, para una mejor comprensión.

Tabla 1. Acrónimos

Acrónimo	Significado	Traducción
AL	Assurance Level	Nivel de Aseguramiento
AVeMaCS	Asisted Verification Management for Critical Systems	Gestión de la Verificación Asistida para sistemas críticos
CENELEC	Comité Européen de Normalisation Electrotechnique	Comité Europeo de Normalización Electrotécnica
CMMi	Capability Maturity Model Integration	Integración de modelos de madurez de capacidades
CSCI	Computer Software Configuration Item	Elemento de Configuración Software
CSV	Comma Separated Values	Valores separados por comas
DAL	Development Assurance Level	Nivel de Aseguramiento para el desarrollo
DCP	Display Control Panel	Panel de Control de Presentaciones
DTU	Data Transfer Unit	Unidad de Transferencia de Datos
DU	Display Unit	Unidad de Presentación
EFIS	Electronic Flight Instrument System	Sistema de Instrumentación Electrónica de Vuelo
EIA	Electronic Industries Alliance	Alianza de Industrias Electrónicas
ETI	Elapsed Time Indicator	Indicador de tiempo transcurrido
ESARR	Eurocontrol Safety Regulatory Requirements	Requisitos Reglamentarios de Seguridad Eurocontrol
FHA	Functional Hazard Analysis	Análisis Funcinal de Amenazas

Tabla 1. Acrónimos

Acrónimo	Significado	Traducción
IEC	International Electrotechnical Commission	Comisión Electrotécnica Internacional
IEEE	Institute of Electrical and Electronics Engineers	Instituto de Ingenieros Eléctricos y Electrónicos
IRS	Interface Requirements Specification	Especificación de Requisitos de Interfaz
MC	Mission Computer	Computadora de Misión
MIL-STD	Military Defense Standard	Estándar Militar de Defensa (Estados Unidos)
MLV	Memory Loader Verifier	Verificador Cargador de Memoria
N/A	No Aplica	N/A
OFP	Operational Flight Program	Programa operacional de vuelo
PFD	Primary Flight Display	Presentación Primaria de Vuelo
RTCA	Radio Technical Commission for Aeronautics	Comisión radio-técnica para la aeronáutica
SAS	Software Accomplishment Summary	Informe de Cumplimiento Software
SCI	Software Configuration Index	Índice de Configuración Software
SCMP	Software Configuration Management Plan	Plan de Gestión de la Configuración Software
SCMR	Software Configuration Management Records	Registros de gestión de la configuración software
SDD	Software Design Document	Documento de Diseño Software
SDP	Software Development Plan	Plan de Desarrollo Software
SECI	Software Life-cycle Configuration Index	Índice de Configuración del entorno del ciclo de vida software
SFD	Secondary Flight Display	Presentación Secundaria de Vuelo
SICGV	Sistema Integrado de Control y Gestión de Vuelo	N/A
SQAP	Software Quality Assurance Plan	Plan de Calidad Software
SQAR	Software Quality Assurance Records	Registros de Aseguramiento de Calidad Software
SRD	Software Requirements Document	Documento de Requisitos Software
SVCP	Software Verification Cases and Procedures	Casos y Procedimientos de Verificación Software

Tabla 1. Acrónimos

Acrónimo	Significado	Traducción
SVP	Software Verification Plan	Plan de Verificación Software
SVR	Software Verification Results	Resultados de la Verificación Software
HTML	HyperText Markup Language	Lenguaje de marcado de hipertexto
PHP	PHP Hypertext Pre-processor	Pre-Procesador de Hipertexto PHP
PSAC	Plan for Software Aspects of Certification	Plan de Certificación Software

CAPÍTULO 2: NORMATIVAS Y PROCESOS

En este capítulo se presentarán las normativas que se pretenden cubrir con esta herramienta, así como los procesos de ingeniería cubiertos dentro del ciclo de vida software.

2.1. MARCO REGULATORIO

Existen varias normativas que se encargan de definir procesos para el desarrollo de software en sistemas críticos. Algunas de ellas son las siguientes:

- ESARR6 (*Eurocontrol Safety Regulatory Requirement 6 – Software in ATM Functional Systems*, Ref. [7]): Aplicable a Sistemas Aeronáuticos no embarcados. La ESARR6 (Ref. [7]) es una continuación del marco definido por la ESARR4 (*Eurocontrol Safety Regulatory Requirement 4 – Risk Assessment and Mitigation in ATM*, Ref.[8]) acerca de los aspectos software. Trata de la implementación de sistemas que aseguren que los riesgos asociados con el uso de sistemas aeronáuticos terrestres se reduzcan hasta un nivel tolerable. Este marco regulatorio no establece ningún estándar de aseguramiento de la seguridad como medida de cumplimiento aceptable. Un estándar reconocido internacionalmente para cumplimiento con esta norma es la DO-278 (Ref. [5]) o la ED-153 (*Guidelines for ANS Software Safety Assurance*, Ref. [9]).

- DO-178B (*Software Considerations in Airborne Systems and Equipment Certification*, Ref. [4]): Sistemas aeronáuticos embarcados. Es el estándar internacional más ampliamente reconocido. Ha servido como base a otros estándares, como son el DO-278 (Ref. [5]). Esta edición ha sido actualizado recientemente, dando lugar a la DO-178C (Ref. [10]).

- DO-278 (*Guidelines for Communication, Navigation Surveillance, and Air Traffic Management*, Ref. [5]): Sistemas aeronáuticos no embarcados. Recientemente ha sido actualizado en la DO-278A (Ref. [11]).

- IEC 61508 (*Functional safety of electrical/electronic/programmable electronic safety-related systems*, Ref. [1]): Automatización en la industria. Este estándar también ha servido de base a varios documentos específicos, como por ejemplo la industria ferroviaria o la de automóviles.

- CENELEC 50128 (*Railway applications - Communications, signalling and processing systems*, Ref. [12]): Es el estándar seguido en el ámbito ferroviario.

- ISO 26262 (*Road Vehicles – Functional Safety*, Ref. [13]): Se refiere a los sistemas de seguridad en automóviles. Tiene como objetivo garantizar la seguridad funcional de un sistema eléctrico/electrónico de un vehículo motor. Se deriva de la norma IEC 61508 (Ref. [1]) para su uso específico en el sector del automóvil.

- IEC 62304 (*Medical device software – Software life cycle processes*, Ref. [14]): Especifica los requisitos del ciclo de vida de desarrollo software en dispositivos médicos.

2.2. NORMATIVAS IMPLEMENTADAS EN LA HERRAMIENTA

La herramienta pretende cubrir los procesos de verificación de la normativa principal en el ámbito de la aeronáutica: la DO-178B (Ref. [4]). Además, la cobertura de los objetivos de la DO-178B (Ref. [4]) serviría de base para que AVeMaCS también pudiese cubrir con los objetivos de la ED-109.

Estas normativas se utilizan en programas con desarrollo software que deben superar niveles importantes en normativas relativas al cumplimiento de requisitos de seguridad física (*safety*).

2.2.1. DO-178B

La DO-178B (*Software Considerations in Airborne Systems and Equipement Certification*, Ref. [4] es el documento más utilizado en los procesos de certificación software. La mayoría del resto de documentos están basados en este, así que representa la opinión consensuada de la industria para la creación de software seguro.

Hay que tener en cuenta que, aunque la DO-178B (Ref. [4]) no habla de metodologías específicas de desarrollo o actividades de gestión, sí que hace énfasis de que siguiendo unos procesos rigurosos, se consiguen beneficios en los costes y en la planificación.

Las actividades de verificación especificadas en la DO-178B (Ref. [4]) son particularmente efectivas identificando problemas del software en las fases tempranas de desarrollo.

Este documento proporciona requisitos reguladores de seguridad para el uso de software en sistemas embarcados. Su objetivo es asegurar que los riesgos asociados a la utilización de software en los sistemas se han reducido a un nivel tolerable.

La DO-178B es un estándar maduro, ya que existe desde hace más de 20 años y ha pasado por distintas revisiones (las previas eran DO-178 y DO-178A). Es un documento de consenso que incluye el punto de vista tanto de la industria como de las autoridades certificadoras. La DO-178B (Ref. [4]) es autocontenida, y no hace referencia a otros estándares software, excepto a los que produzca el desarrollador para cumplir con algunos de los objetivos de la misma.

Existen comparativas entre la DO-178B (Ref. [4]) y otros estándares software, como la MIL-STD-498 (Ref. [15]), MIL-STD-2167A (Ref. [16]), IEEE/EIA-12207 (Ref. [17]), IEC 61508 (Ref. [12]) y el estándar de defensa del reino unido 00-55 (Ref. [18]). No hay ninguno de ellos que cubra todos los objetivos del los objetivos de la DO-178B. Además, estos otros estándares carecen de un criterio en cuanto a los objetivos y los análisis de safety están enfocados a nivel de sistema. No obstante, la experiencia en la aplicación de estos estándares normalmente facilita el camino para adoptar la DO-178B.

La AC-20-115B (Ref. [19]) habla de la DO-178B (Ref. [4]) como un medio aceptable, pero no el único medio, para recibir la aprobación regulatoria de software en sistemas que van a ser certificados bajo una autorización TSO, un certificado de tipo (TC) o un suplemento al certificado de Tipo (STC).

Muchos de los desarrolladores utilizan la DO-178B (Ref. [4]) para evitar el trabajo que implica mostrar que otros medios de cumplimiento son equivalentes a la DO-178B (Ref. [4]). Aunque la DO-178B (Ref. [4]) ha sido escrita como guía, se ha convertido en una práctica estándar dentro de la industria. La DO-178B (Ref. [4]) está oficialmente reconocida como un estándar internacional de facto por la ISO (Organización Internacional de Estandarización).

La DO-178B (Ref. [4]) utiliza el mecanismo de Niveles de Garantía de Seguridad (DAL, *Development Assurance Level*).

Este mecanismo permite conocer el nivel necesario de profundidad en el análisis del elemento considerado, así como el nivel de validación y verificación; junto con el grado de exigencia las evidencias a recolectar.

De esta forma, se permite una asignación simple de los Requisitos de Seguridad del Sistema, que pueden asociarse a un nivel de garantía desarrollado, y a evidencias asociadas a proporcionar la garantía necesaria de que un componente particular del Sistema es seguro.

La DO-178B (Ref. [4]) define cinco niveles de criticidad (A, B, C, D y E), para los cuales existen unos objetivos y actividades que hay que satisfacer con un grado de independencia.

Tabla 2. Objetivos de la DO-178B por nivel

Nivel	Condición de Fallo	Objetivos	Con independencia
A	Catastrófica	66	25
B	Severa	65	14
C	Mayor	57	2
D	Menor	28	2
E	Sin Efecto	0	0

Estos objetivos están definidos según distintos procesos, que son:

- Planificación
- Desarrollo
- Requisitos
- Diseño
- Implementación
- Integración
- Verificación
- Gestión de la Configuración Software
- Calidad Software

Cada uno de estos procesos tiene entradas, salidas y un criterio de transición entre procesos.

Para demostrar el cumplimiento de cada uno de los objetivos aplicables, es necesario proporcionar unas evidencias.

Por ejemplo, para el objetivo “¿Es verificable el código fuente?” sería necesario apoyarse en análisis o pruebas que demuestren que el código fuente no contiene estructuras que no pueden ser probadas.

Por lo tanto, es muy importante que todas las actividades del ciclo de vida estén trazadas con los objetivos aplicables. En los siguientes apartados veremos los objetivos para cada uno de los procesos descritos en la norma.

2.2.1.1 Proceso de Planificación

El proceso de planificación, como el resto de procesos, define unos objetivos que dependen del nivel del software. Estos objetivos están definidos en la Tabla A-1 de la DO-178B (Ref [4]).

El proceso de planificación define cinco planes y tres estándares, cuyos contenidos mínimos están especificados dentro de la normativa. Estos planes y estándares son los siguientes:

- **Plan de Certificación Software (PSAC)**

Este plan es el documento que la Autoridad Certificadora utiliza para determinar si el desarrollador está proponiendo un ciclo de vida que tenga en cuenta y vaya a seguir todos los procesos y consideraciones que se definen en la normativa con el suficiente rigor.

- **Plan de Desarrollo Software (SDP)**

Este plan debe incluir los objetivos, estándares y ciclo de vida que se va a utilizar en el proceso de desarrollo software. Puede estar incluido en el PSAC, aunque normalmente es un documento independiente.

- **Plan de Verificación Software (SVP)**

Este plan es una descripción de los procesos de verificación para satisfacer los objetivos del proceso de verificación. Estos objetivos dependen del nivel de software requerido.

- **Plan de Configuración Software (SCMP)**

Este plan establece los métodos que se van a utilizar para cumplir con los objetivos del proceso de gestión de la configuración a través del ciclo de vida software.

- **Plan de Calidad Software (SQAP)**

El Plan de Calidad Software establece los métodos que se van a utilizar para conseguir los objetivos del proceso de evaluación de la Calidad del software. El plan de Calidad Software puede incluir descripciones de mejora de procesos, métricas y otros métodos.

- **Estándar de Requisitos Software**

Este estándar debe definir los métodos, reglas y herramientas que se van a utilizar para desarrollar los requisitos de alto nivel.

- **Estándar de Diseño Software**

En este documento se deben definir los métodos, reglas y herramientas que se van a utilizar para desarrollar la arquitectura software y los requisitos de bajo nivel.

- **Estándar de Codificación Software**

El propósito de este estándar es definir los lenguajes de programación, métodos, reglas y herramientas para la codificación del software.

2.2.1.2. Proceso de Desarrollo

Los objetivos del proceso de desarrollo software están definidos en la Tabla A-2 de la DO-178B (Ref [4]).

Como se puede comprobar observando la tabla, estos objetivos son aplicables para todos los niveles, ya que el proceso de desarrollo se debe llevar a cabo independientemente del nivel de seguridad aplicable al software.

Debido a que estos objetivos se refieren al proceso de desarrollo en general, es posible integrar estos objetivos en otros procesos. Por ejemplo, los objetivos 1 y 2 de la Tabla A-2

pueden formar parte de los objetivos del proceso de requisitos, ya que están estrechamente ligados.

2.2.1.3. Proceso de Requisitos Software

Los objetivos del proceso de requisitos software están definidos en la Tabla A-3 de la DO-178B, Ref. [4].

Como hemos visto en el proceso de desarrollo, deben existir requisitos software, independientemente del nivel requerido.

Ahora, en el proceso de requisitos software, hay tres objetivos principales:

- Los requisitos software deben cumplir con los requisitos de sistema
- La definición de los requisitos software debe ser precisa y consistente.
- Los requisitos software deben estar trazados a los requisitos de sistema

De esta forma es posible asegurar que los requisitos de sistema que son aplicables al software están perfectamente trasladados a la definición de los requisitos software, y que esta definición está correctamente realizada para que no existan malinterpretaciones o falte información.

Para niveles más altos, se requiere otro tipo de evidencias, como que los requisitos sean verificables o evidencias de cumplimiento con los estándares definidos.

2.2.1.4. Proceso de Diseño Software

Los objetivos del proceso de diseño software están definidos en la Tabla A-4 de la DO-178B, Ref [4].

Los objetivos del proceso de diseño dependen fuertemente del nivel requerido. De hecho, para un DAL-D, sólo es necesario confirmar la integridad de las particiones. Para niveles más altos, es necesario aportar evidencias de cumplimiento de los requisitos de bajo nivel con los de alto nivel, de cumplimiento con los estándares y de la arquitectura software. En el nivel más alto, además de aplicar independencia para algunos objetivos, hay que demostrar otros objetivos, como la precisión de los algoritmos, o la compatibilidad de la arquitectura software con el hardware.

Si el código fuente es generado directamente desde los requisitos de alto nivel, entonces los requisitos de alto nivel también pueden ser considerados requisitos de bajo nivel, por lo que aplicarían también estos objetivos sobre esos requisitos. En la práctica, este planteamiento puede depender de varios factores, como la complejidad del software a desarrollar, el nivel requerido o incluso la Autoridad Certificadora.

2.2.1.5. Proceso de Implementación

Los objetivos del proceso de implementación software están definidos en la Tabla A-5 de la DO-178B, Ref. [4].

Para el proceso de implementación no se piden evidencias para un nivel D. Las evidencias para un nivel C son las mismas que para niveles superiores, excepto que no se necesita presentar evidencias de cumplimiento con el estándar definido.

Entre las evidencias de cumplimiento con los objetivos de este proceso está la trazabilidad con los requisitos de bajo nivel y el cumplimiento con los estándares definidos y la arquitectura.

2.2.1.6. Proceso de Integración

Los objetivos del proceso de integración software están definidos en la Tabla A-6 de la DO-178B, Ref. [4].

En el proceso de integración deben presentarse evidencias de cumplimiento independientemente del nivel. Para un nivel D, debe demostrarse que el código objeto ejecutable cumple y además es robusto con los requisitos de alto nivel, y que es compatible con el hardware. Para niveles mayores, debe evidenciarse además el cumplimiento y la robustez con los requisitos de bajo nivel.

2.2.1.7. Proceso de Verificación de la Verificación

Los objetivos del proceso de Verificación de la Verificación software están definidos en la Tabla A-7 de la DO-178B, Ref. [4].

Este proceso tiene como misión principal la revisión de que el proceso de verificación se ha llevado a cabo correctamente. Aunque para un nivel D sólo es necesario evidenciar que la cobertura de pruebas de los requisitos de alto nivel ha sido alcanzada correctamente. Para niveles más altos, es necesario evaluar la corrección de los procedimientos de pruebas y analizar la cobertura estructural, entre otros objetivos.

2.2.1.8. Proceso de Gestión de la Configuración Software

Los objetivos del proceso de Gestión de la Configuración Software están definidos en la Tabla A-8 de la DO-178B (Ref [4]).

El proceso de gestión de la configuración es uno de los procesos integrales, que debe llevarse a cabo siempre independientemente del nivel requerido. Los objetivos a conseguir son los mismos para cualquier nivel.

2.2.1.9. Proceso de Gestión de la Calidad Software

Los objetivos del proceso de Gestión de la Calidad Software están definidos en la Tabla A-9 de la DO-178B (Ref [4]).

El proceso de Gestión de la Calidad también es un proceso integral. En este caso, los objetivos sí que son distintos dependiendo del nivel. En cualquier caso siempre hay que llevar a cabo la revisión de conformidad del software, donde se evalúa que se han completado los procesos del ciclo de vida, los datos del ciclo de vida son completos y que el código objeto ejecutable está controlado y puede se regenerado.

2.2.2. DO-278

La DO-278 (*Guidelines for Communication, Navigation Surveillance, and Air Traffic Management (CNS/ATM) Systems Software Integrity Assurance*, Ref. [5]), también conocida como ED-109, es un documento aplicable a los sistemas no embarcados. Está basado en la DO-178B (Ref. [4]]), y pretende ser una interpretación de ésta para los sistemas no embarcados.

2.3. PROCESOS DE INGENIERÍA

Para el desarrollo del software, puede tomarse cualquier modelo para el ciclo de vida, como pueden ser el desarrollo en cascada o el desarrollo en V. En los proyectos de desarrollo de software crítico, el modelo más utilizado es el modelo en "V", que se muestra en la siguiente figura:

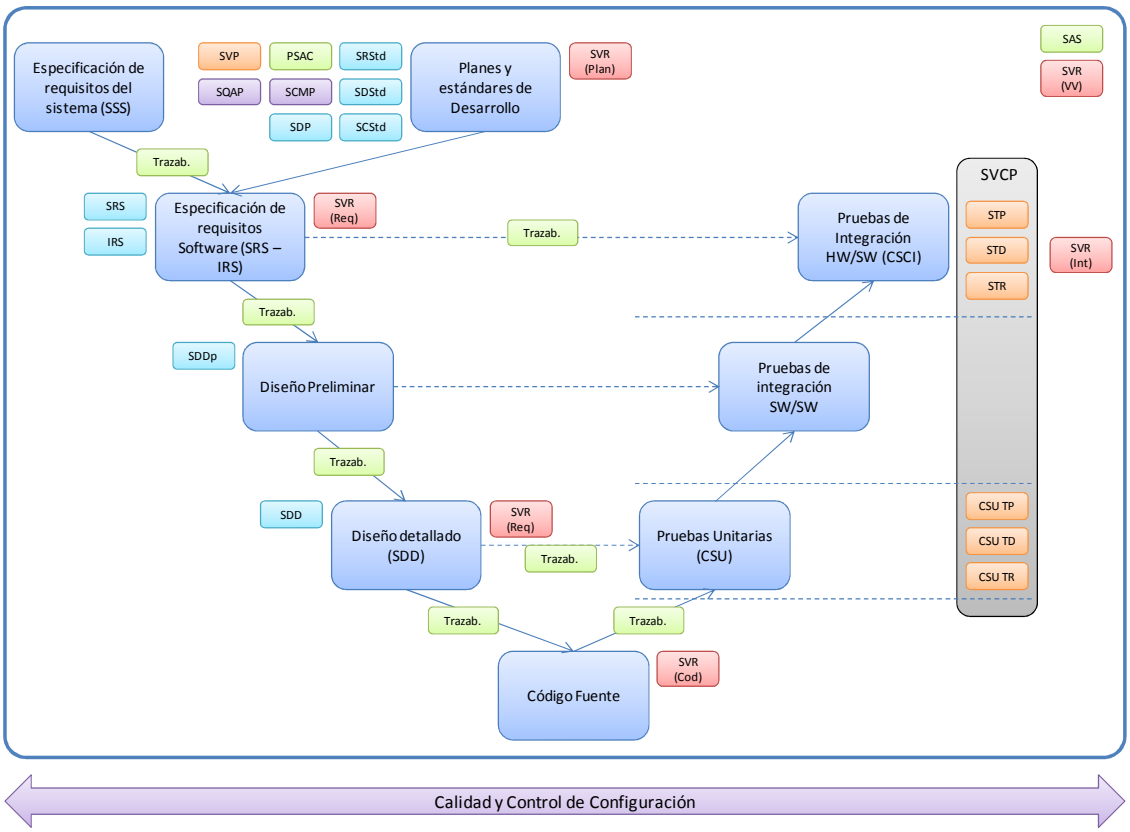
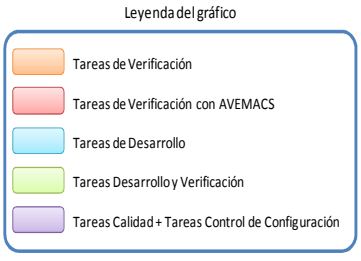


Figura 6. Ciclo de vida en "V"



Según esta figura, se puede agrupar el ciclo de vida en cinco procesos diferentes, que cumplen con las normativas aplicables:

1. Proceso de planificación software. El proceso de planificación tiene como objetivo determinar el conjunto de tareas que deben ser realizadas para producir un software seguro. Los documentos a generar serían los siguientes:

- Plan de Aspectos Software de Certificación (PSAC). Este documento proporciona detalles sobre los requisitos del producto y sobre la forma en la que se van a cumplir dichos requisitos.
- Plan de Desarrollo Software (SDP). Define el ciclo de vida software y el entorno de desarrollo.
- Plan de Verificación Software (SVP). Define el método propuesto de verificación que satisface todos los objetivos del proceso de verificación del software.
- Plan de Gestión de la Configuración Software (SCMP). Define el método propuesto de gestión de la configuración software.
- Plan de Aseguramiento de la Calidad Software (SQAP). Contiene el plan propuesto para satisfacer los objetivos del proceso de aseguramiento de la calidad software.

Adicionalmente y dependiendo del nivel, puede ser necesario generar los siguientes estándares:

- Estándar de Requisitos Software
- Estándar de Diseño Software
- Estándar de Codificación

2. Desarrollo Software: El proceso de desarrollo software se descompone en 4 subprocesos:
   - Requisitos Software. Define los requisitos de alto nivel (funcionales, operacionales, de interfaz, de seguridad, etc.) que aplican al software.
   - Diseño Software. Requisitos de bajo nivel utilizados para implementar el código fuente.
   - Codificación. Producción del código fuente a partir del diseño.
   - Integración. Integración del código en el entorno de ejecución.

   El proceso de desarrollo software genera las siguientes salidas:
   - Requisitos de Software (SRD)
   - Descripción del Diseño Software (SDD)
   - Código fuente
   - Código Objeto Ejecutable
3. Verificación Software. El propósito es identificar y reportar cualquier error resultante del proceso de desarrollo. Los objetivos del proceso de verificación puede ser alcanzados mediante revisiones, walkthroughs, unit testing, integration testing, etc.

   El proceso de verificación genera las siguientes salidas:
   - Casos y Procedimientos de Verificación Software (SVCP)
   - Resultados de la Verificación Software (SVR)

4. Gestión de la Configuración Software. El propósito es establecer un control de configuración seguro y efectivo de todos los elementos producidos a lo largo del ciclo de vida software. Las siguientes actividades deben ser realizadas:

   - Identificación de la Configuración. Todos los elementos bajo control de configuración deben ser unívocamente identificados y referenciados.
   - Control de Cambios. Hay que establecer un mecanismo de control de cambios para los elementos bajo control de configuración. Los cambios tienen que ser trazables y reversibles.
   - Establecimiento de la Línea de Referencia. Una línea de referencia, o punto de origen, debe ser definido para cada elemento bajo control de configuración.
   - Archivado del software. El software debe ser periódicamente archivado (backups).

   El proceso de configuración genera las siguientes salidas:

   - Índice de Configuración del Entorno del Ciclo de Vida Software (SECI)
   - Índice de Configuración Software (SCI)
   - Informes de incidencias
   - Registros de Configuración Software (SCMR)

5. Aseguramiento de la Calidad Software. El objetivo de este proceso es asegurar que el ciclo de vida software va a producir un software de calidad. Esto se lleva a cabo revisando las transiciones entre procesos para comprobar que las salidas de un proceso se adaptan a lo esperado y que son aptas para ser utilizadas como entradas para el proceso siguiente. Cualquier cambio en los planes originalmente propuestos debe ser evaluado y resuelto para asegurar la consistencia del proceso.

   El proceso de aseguramiento de la calidad genera las siguientes salidas:

   - Registros de Aseguramiento de Calidad (SQAR)
   - Informe de Cumplimiento Software (SAS)

La herramienta AVeMaCS cubre el proceso de verificación software dentro del ciclo de vida de desarrollo software. El proceso de verificación es un proceso integral. La forma de cubrir las tareas se detalla en los siguientes sub-apartados.

La herramienta permite crear los registros de verificación de los elementos de configuración identificados. En el caso de que sea necesaria la creación de un documento de resultados de la verificación, estos registros sirven como base de dicha creación.

En los siguientes sub-apartados veremos las listas de comprobación creadas por defecto en la herramienta. Estas listas de comprobación se presentan en unas tablas, cuyas columnas tienen el siguiente significado:

- Objetivo: Pregunta para cubrir un objetivo concreto de la normativa aplicable.
- Objetivo de la tabla: Referencia a la tabla y número de objetivo de la normativa aplicable.
- Referencia [Normativa Aplicable]: Párrafo de la normativa donde se pueden encontrar más detalles acerca del objetivo a cumplir.
- Nivel mínimo aplicable: Nivel mínimo para el cual el objetivo es aplicable.

Aunque se ha tomado como referencia la normativa aplicable para los contenidos mínimos, existen estándares internacionales reconocidos para prácticamente todos los documentos a realizar y actividades a llevar a cabo, como por ejemplo los proporcionados por el IEEE. Estos estándares también se pueden utilizar como referencia y en muchos de los casos incluyen la información necesaria para cumplir con la normativa.

2.3.1. Proceso de Planificación

El proceso de planificación implica la creación de cinco planes y tres estándares. Estos documentos se han introducido como documentos por defecto en la herramienta, incluyendo unas listas de comprobación definidas para cada uno de ellos. Las listas de comprobación introducidas por defecto se presentan en las siguientes tablas:

Tabla 3. Lista de comprobación del proceso de Planificación

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Están definidos los procesos de desarrollo software y las actividades de los procesos integrales de acuerdo al apartado 4.1a de la DO-178B?	A-1, #1	4.1a	DAL-D
¿Están definidos los procesos de desarrollo software y las actividades de los procesos integrales de acuerdo al apartado 4.3 de la DO-178B?	A-1, #1	4.3	DAL-C
¿Está definido el criterio de transición, las interrelaciones y secuenciación entre procesos de acuerdo al párrafo 4.1b de la DO-178B?	A-1, #2	4.1b	DAL-C
¿Está definido el entorno del ciclo de vida de acuerdo al párrafo 4.1c de la DO-178B?	A-1, #3	4.1c	DAL-C
¿Se han tenido en cuenta las consideraciones adicionales de acuerdo al párrafo 4.1d de la DO-178B?	A-1, #4	4.1d	DAL-D
¿Están definidos los estándares de desarrollo software?	A-1, #5	4.1e	DAL-C
¿Cumplen los planes software con este documento?	A-1, #6	4.1f, 4.6	DAL-C
¿Están coordinados los planes?	A-1, #7	4.1g, 4.6	DAL-C

En las listas de comprobación incluidas para cada uno de los planes, se toma como referencia los contenidos que aparecen en la normativa. Lo mínimo sería incluir una pregunta por cada uno de los objetivos o contenidos aplicables, pero se pueden incluir más preguntas si se quiere añadir algún matiz o aumentar el detalle en el registro de las respuestas.

En el caso del PSAC, es importante que haya definida una visión general del sistema y del software dentro del sistema, ya que es importante situar en contexto el software que va a ser objeto de la certificación. Hablando de este aspecto, también es necesario hablar de las bases de certificación que se van a usar, el nivel de seguridad requerido y la descripción de las actividades a realizar. Se debe incluir también una descripción del ciclo de vida, las actividades

que se van a llevar a cabo, los datos y documentos que se van a generar que cubren esas actividades, y la planificación dentro del proyecto.

Un apartado también necesario es el de las “consideraciones adicionales”. En este apartado del plan se debe explicar las características que puedan afectar al proceso de certificación, tal como métodos alternativos de cumplimiento, calificación de herramientas, si hay software desarrollado con anterioridad que se vaya a utilizar en el proyecto, si hay software COTS y cualquier otra cosa similar. Siempre es recomendable incluir el apartado conteniendo como mínimo las consideraciones nombradas en la normativa, y en el caso de que no apliquen al proyecto, indicarlo expresamente. De esta forma, la Autoridad puede tener evidencia de que no se ha olvidado tenerlas en cuenta para el proyecto, sino que simplemente no aplica en el caso concreto.

Tabla 4. Lista de comprobación del PSAC

Objetivo	Referencia DO-178B
Visión general del Sistema	11.1a
¿Existe y está completa la información acerca de la visión general del sistema? Esta información debe incluir la asignación de funcionalidades que aplican al hardware y al software, la arquitectura, los procesadores utilizados, los interfaces hardware/software, y las consideraciones de safety. Si el sistema es únicamente software, se debe indicar "No aplicable. Este proyecto es para un componente de sistema"	
Visión general del software	11.1b
¿Existe y está completa la información acerca de la visión general del software? Esta información debe incluir las consideraciones de safety para el software, y otras consideraciones como compartición de recursos, redundancia, software disímil multiversión, tolerancia ante fallos y estrategias de planificación y temporización.	
Consideraciones para la certificación	11.1c
¿Está identificado y justificado el nivel de safety del software?	
¿Hay una descripción de las actividades de certificación para cada uno de los siguientes elementos? - Documentación de requisitos - Plan de Verificación - Plan de configuración y Plan de Calidad - Procedimientos de pruebas - Resultados de las pruebas - Matrices de trazabilidad - Índice de Configuración Software (SCI) - Índice de Cumplimiento Software (SAS) - Métodos alternativos de cumplimiento	
Ciclo de Vida Software	11.1d
¿Está definido el ciclo de vida software?	
Datos del ciclo de vida	11.1e
¿Están definidos los datos del ciclo de vida que van a ser producidos?	
¿Están los datos del ciclo de vida correctamente relacionados con las actividades definidas?	
Planificación	11.1f

Tabla 4. Lista de comprobación del PSAC

Objetivo	Referencia DO-178B
¿Está definida la planificación de las actividades del ciclo de vida de desarrollo?	
Consideraciones adicionales	11.1g
¿Se han tenido en cuenta las siguientes consideraciones adicionales? - Métodos alternativos de cumplimiento - Calificación de herramientas - Software previamente desarrollado - Software COTS - Software Disímil Multiversión - Información de historia en servicio	

El Plan de Desarrollo Software (SDP) tiene que definir perfectamente cómo se va a desarrollar el mismo. Empezando por la definición de estándares (especialmente si es necesario por el nivel aplicable) y los procesos del ciclo de vida. También es necesario definir el entorno de desarrollo, las herramientas para los requisitos, el diseño y codificación.

Tabla 5. Lista de comprobación del SDP

Objetivo	Referencia DO-178B
Estándares	11.2a
¿Se han identificado los estándares de requisitos, diseño y codificación?	
¿Se han identificado los estándares para el software previamente desarrollado, incluyendo los COTS?	
Ciclo de Vida Software	11.2b
¿Se han definido los procesos del ciclo de vida software?	
¿Se ha definido el criterio de transición?	
Entorno de desarrollo software	11.2c
¿Se ha definido el entorno de desarrollo software?	
¿Se ha definido el método y herramientas para el desarrollo de requisitos?	
¿Se ha definido el método y herramientas para el diseño?	
¿Se ha definido el lenguaje de programación, las herramientas de desarrollo y el compilador a utilizar?	
¿Se ha identificado la plataforma hardware que se va a utilizar?	

El Plan de Verificación Software (SVP) definirá las actividades de verificación que se llevarán a cabo en el proyecto. Debido a que para algunos niveles de seguridad es necesaria la

independencia entre las tareas de desarrollo y verificación, es necesario definir la organización y la independencia de los distintos equipos participantes en el proyecto. Los métodos de verificación participarán en la estrategia definida para las actividades de verificación necesarias.

Siempre es necesario definir el entorno y herramientas de verificación, aunque esta información puede estar contenida en otros documentos (como por ejemplo los planes de pruebas) y referenciados en este plan.

Hay una serie de consideraciones específicas que hay que tener en cuenta a la hora de verificar en el caso de que sean aplicables para este proyecto. Estas consideraciones son, por ejemplo, la existencia y verificación de las particiones, las suposiciones acerca del compilador, la existencia de software previamente desarrollado y el software disímil multiversión.

Una cosa importante a tener en cuenta es la estrategia de re-verificación. En el caso de que alguno de los productos software ya verificados sufra un cambio, es de utilidad especificar si se va a repetir la verificación, o si por el contrario se va a realizar un análisis de impacto y se va a verificar lo afectado por el cambio. Cualquier otra estrategia o consideración se debe indicar en el Plan de Verificación Software.

Tabla 6. Lista de comprobación del SVP

Objetivo	Referencia DO-178B
Organización	11.3a
¿Está definida la organización dentro del proceso de verificación y los interfaces con el resto de los procesos del ciclo de vida?	
Independencia	11.3b
¿Está definida la independencia del proceso de verificación (si es necesaria)?	
¿Es suficiente el nivel de dependencia especificada para los procesos de verificación?	
Métodos de Verificación	11.3c
¿Están definidos los métodos de verificación para cada una de las actividades del proceso de verificación?	
Entorno de Verificación	11.3d
¿Están definidos los equipos de pruebas, herramientas de análisis y herramientas para las pruebas que se van a utilizar? En caso de que alguno de estos elementos esté descrito en otro documento, deberá aparecer una referencia al mismo.	
¿Está definido el entorno hardware de pruebas? En caso de que el entorno se defina en otro documento, debe aparecer una referencia al mismo.	
Criterio de Transición	11.3e
¿Está definido el criterio de transición para llevar a cabo las tareas de verificación?	
Consideraciones acerca de las particiones	11.3f
En el caso de que existan particiones, ¿están definidos los métodos para verificar la integridad de las mismas?	

Tabla 6. Lista de comprobación del SVP

Objetivo	Referencia DO-178B
Suposiciones acerca del compilador	11.3g
¿Existe una descripción de las suposiciones acerca del compilador?	
Estrategia de re-verificación	11.3h
¿Está definida la estrategia de re-verificación cuando existan modificaciones en el software?	
Software previamente desarrollado	11.3i
En el caso de que exista software previamente desarrollado que no cumpla los objetivos de la normativa aplicable, ¿está descrita la forma en la que se van a satisfacer los objetivos de la normativa aplicable?	
Software disímil multiversión	11.3j
Si se utiliza la técnica de software disímil multiversión, ¿Están descritos los procesos de verificación en este caso?	

Otro de los procesos integrales se define en el Plan de Gestión de la Configuración Software (SCMP). Tiene mucha importancia el hecho de que los cambios estén totalmente controlados a lo largo del ciclo de vida del software. Cualquier cambio o incidencia debe ser registrado y trazado a los productos afectados.

Para ello es necesario definir el entorno de configuración: herramientas, procedimientos, métodos, organización, responsabilidades e interfaces entre los procesos.

Todas las actividades deben estar descritas: identificación de la configuración, establecimiento de líneas base, informes de incidencias, control de cambios, seguimiento del estado de la configuración, etc.

La lista de comprobación que se presenta a continuación está detallada en cuanto a la información que es necesario incluir en el SCMP.

Tabla 7. Lista de comprobación del SCMP

Objetivo	Referencia DO-178B
Entorno	11.4a
¿Está definido el entorno de configuración software definido? Esto incluye herramientas, procedimientos, métodos, estándares, organización, responsabilidades y los interfaces entre los procesos definidos.	
Actividades	11.4b
¿Existe una descripción para cada una de las actividades del proceso de configuración software de los siguientes?	
¿Está definida la identificación de la configuración? Elementos a ser identificados, cuándo van a ser identificados, los métodos de identificación para los elementos de configuración y la relación de la identificación con el sistema.	
¿Están descritas las líneas base y la trazabilidad? Cómo se establecen las líneas base, cuándo serán establecidas, y la trazabilidad de las líneas base con los elementos de configuración.	

Tabla 7. Lista de comprobación del SCMP

Objetivo	Referencia DO-178B
¿Está descrito el método para los informes de problemas? Contenido e identificación de los mismos, cuándo son escritos, criterio y método para cerrarlos, y su relación con la actividad de control de cambios.	
¿Está descrita la actividad de control de cambios? Elementos de configuración y líneas base a ser controlados, cuándo serán controlados y métodos para preservar la integridad de las líneas base y los elementos de configuración.	
¿Está descrito el método para la revisión de cambios?	
¿Está descrito el método para llevar a cabo el seguimiento del estado de la configuración?	
¿Están descritas las actividades de archivo, recuperación y entrega de la información bajo control de configuración?	
¿Están definidos los mecanismos de control de carga?	
¿Están definidos los controles para las herramientas de desarrollo y verificación?	
¿Están definidos los controles asociados a las categorías de control para los elementos de configuración?	
Criterio de Transición	11.4c
¿Está definido el criterio de transición para los procesos de control de configuración?	
Datos de configuración	11.4d
¿Están definidos los informes de configuración a generar? Esto incluye SCI, SECI y registros de configuración.	
Control de Proveedores	11.4e
¿Se han definido los métodos para asegurar que los sub-contratistas cumplirán con el Plan de Configuración Software?	

La gestión de la Calidad es otro de los procesos integrales definidos por la norma. Como hemos visto en los objetivos para el proceso de Calidad, la responsabilidad de estas actividades debe ser independiente de las actividades del proceso de desarrollo.

Para ello, es necesario definir el entorno para estas actividades: alcance, responsabilidades, estándares, procedimientos, herramientas y métodos.

Las actividades a realizar deben ser descritas y controladas a lo largo de todo el ciclo de vida. Además, se deben guardar registros de los resultados obtenidos en su realización, que sirven como evidencia para demostrar su ejecución.

La “Software Conformity Review” o “Informe de Cumplimiento del Software” es una actividad imprescindible que se utiliza para evidenciar que se han completado los procesos del ciclo de vida software, se han completado los productos del ciclo de vida software y que el código objeto ejecutable está controlado y puede ser regenerado.

Tabla 8. Lista de comprobación del SQAP

Objetivo	Referencia DO-178B
Entorno	11.5a
¿Está descrito el entorno para las actividades de Calidad? Esto incluye alcance, responsabilidades, estándares, procedimientos, herramientas y métodos.	
Autoridad	11.5b
¿Está establecida la autoridad de Calidad?	
¿Está establecida la independencia y responsabilidad de las actividades de Calidad?	
Actividades	11.5c
¿Se han definido las actividades a llevar a cabo para cada fase del ciclo de vida?	
¿Se han definido los métodos para llevar a cabo las actividades de Calidad dentro del ciclo de vida?	
¿se han definido las actividades a llevar a cabo con respecto a los informes de problemas?	
¿Se ha definido la "Software Conformity Review"?	
Criterio de Transición	11.5d
¿Se ha definido el criterio de transición para las actividades de Calidad?	
Planificación	11.5e
¿Se ha definido la relación temporal de las actividades de Calidad con respecto a las actividades del ciclo de vida software?	
Registros de Calidad	11.5f
¿Se han definido los registros de Calidad que se van a generar?	
Control de Proveedores	11.5g
¿Se han definido los métodos para asegurar que los sub-contratistas cumplirán con el Plan de Calidad?	

El estándar de requisitos software (referenciado como SRStd), debe definir los métodos, la notación, las herramientas y la forma de introducir requisitos derivados.

El estándar de requisitos servirá para que los requisitos estén controlados, se creen de forma homogénea y sirvan para su propósito.

Es altamente recomendable utilizar una herramienta específica de gestión de requisitos, ya que otros métodos de almacenamiento y control de los requisitos pueden hacer que cuando el volumen de los mismos aumente, sea muy difícil la gestión de los mismos.

Tabla 9. Lista de comprobación del SRStd

Objetivo	Referencia DO-178B
Métodos para los requisitos	11.6a
¿Están definidos los métodos para el desarrollo de requisitos?	
Notaciones	11.6b
¿Está definida la notación para expresar los requisitos?	
Uso de herramientas de gestión de requisitos	11.6c
¿Están definidas las herramientas de gestión de los requisitos?	
Método para los requisitos derivados	11.6d
¿Está definido el método para introducir requisitos derivados?	

El estándar de diseño debe incluir la definición de los métodos para describir el diseño, convenciones, condiciones, herramientas y limitaciones. Este estándar servirá para establecer las pautas y criterios a la hora de la realización del diseño por parte del desarrollo.

Tabla 10. Lista de comprobación del SDStd

Objetivo	Referencia DO-178B
Métodos para el diseño	11.7a
¿Se han definido los métodos para describir el diseño?	
Convenciones para los nombres	11.7b
¿Se han definido las convenciones para los nombres?	
Condiciones en los métodos de diseño	11.7c
¿Se han descrito las condiciones impuestas para los métodos de diseño?	
Uso de herramientas de diseño	11.7d
¿Están definidas las herramientas para el diseño?	
Limitaciones para el diseño	11.7e
¿Están definidas las limitaciones para el diseño?	
Restricciones de complejidad	11.7f
¿Están definidas las restricciones de complejidad en el diseño?	

El estándar de codificación debe especificar el lenguaje de programación, las reglas sintácticas, las convenciones en los nombres y las limitaciones en la codificación.

La complejidad ciclomática puede determinar el número de pruebas a realizar para alcanzar ciertos niveles de cobertura estructural. De la misma forma, poner limitaciones en la complejidad y el tamaño de funciones, métodos, ficheros y otros elementos puede ayudar a la comprensión y mantenibilidad del código.

Tabla 11. Lista de comprobación del SCStd

Objetivo	Referencia DO-178B
Lenguaje de programación	11.8a
¿Está definido el lenguaje de programación a utilizar?	
En caso de ser un subconjunto del lenguaje, ¿está claramente definido este subcobjunto?	
Reglas sintácticas	11.8b
¿Están definidas las reglas sintácticas?	
¿Permiten las reglas sintácticas que el código sea homogéneo y comprensible?	
Convenciones de nombres en el código	11.8c
¿Están definidas las convenciones de los nombres para componentes, funciones, variables y constantes?	
Limitaciones en la codificación	11.8d
¿Están descritas las limitaciones en la complejidad del código?	
Herramientas de codificación	11.8e
¿Están definidas las herramientas de codificación?	

2.3.2. Proceso de Desarrollo

El proceso de desarrollo define actividades y productos que deben ser realizados a lo largo del ciclo de vida. Entre ellos está la creación de requisitos software de alto nivel, la arquitectura, requisitos software de bajo nivel y el código fuente.

El seguimiento del proceso de desarrollo da una idea del estado avance del mismo, y de las actividades relacionadas que pueden llevarse a cabo.

Se ha definido la lista de comprobación para el proceso de desarrollo que se detalla en la Tabla 12.

Tabla 12. Lista de comprobación del proceso de Desarrollo

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Han sido desarrollados los requisitos de alto nivel?	A-2, #1	5.1.1a	DAL-D
¿Han sido definidos los requisitos derivados de alto nivel?	A-2, #2	5.1.1b	DAL-D
¿Ha sido desarrollada la arquitectura software?	A-2, #3	5.2.1a	DAL-D
¿Han sido desarrollados los requisitos de bajo nivel?	A-2, #4	5.2.1a	DAL-D
¿Han sido definidos los requisitos derivados de bajo nivel?	A-2, #5	5.2.1b	DAL-D
¿Ha sido desarrollado el código fuente?	A-2, #6	5.3.1a	DAL-D
¿Ha sido producido el código objeto ejecutable y se ha integrado en el hardware de destino?	A-2, #7	5.4.1a	DAL-D

2.3.3. Proceso de Requisitos

El proceso de requisitos se refiere a los requisitos software de alto nivel. Estos requisitos tienen que estar basados en los requisitos de sistema y trazarse a ellos. Otros aspectos que tienen que ser considerados, es que los requisitos sean precisos y consistentes, que sean compatibles con la plataforma hardware o que los requisitos sean verificables.

Se ha definido la lista de comprobación para el proceso de requisitos que se detalla en la Tabla 13.

Tabla 13. Lista de comprobación del proceso de Requisitos

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Los requisitos software de alto nivel cumplen con los requisitos de sistema?	A-3, #1	6.3.1a	DAL-D
¿Los requisitos software de alto nivel son precisos y consistentes?	A-3, #2	6.3.1b	DAL-D
¿Los requisitos software de alto nivel son compatible con la plataforma hardware?	A-3, #3	6.3.1c	DAL-B
¿Los requisitos software de alto nivel son verificables?	A-3, #4	6.3.1d	DAL-C
¿Los requisitos software de alto nivel cumplen con los estándares?	A-3, #5	6.3.1e	DAL-C
¿Los requisitos software de alto nivel están trazados con los requisitos de sistema?	A-3, #6	6.3.1f	DAL-D
¿Son precisos los algoritmos?	A-3, #7	6.3.1g	DAL-C

Uno de los productos del proceso de requisitos es el SRD, donde quedan registrados. Este documento debe reflejar los requisitos software de alto nivel que vienen desde los requisitos de sistema. Se debe incluir las condiciones potenciales de fallo del software, los requisitos funcionales, operacionales, de precisión y de consistencia, entre otros.

También sería necesario describir los interfaces, aunque, si esta información es demasiado extensa, podría ser descrita en un documento aparte, como un IRS. En cualquier caso, los requisitos de este documento adicional deben seguir las mismas consideraciones que el correspondiente SRD.

Si el software se compone de varios CSCIs, es posible tener un SRD por cada uno de ellos, para así diferenciar mejor la información correspondiente a cada uno de ellos.

Tabla 14. Lista de comprobación del SRD

Objetivo	Referencia DO-178B
Asignación de los requisitos de sistema	11.9a
¿Está descrita la asignación de requisitos de sistema al software?	
¿Están descritos los requisitos safety de sistema asignados al software?	
¿Están descritas las condiciones potenciales de fallo?	
Requisitos funcionales y operacionales	11.9b
¿Están descritos los requisitos software funcionales y operacionales para cada modo de operación?	
Requisitos de Precisión	11.9c
¿Están definidos los requisitos de precisión y consistencia?	
Requisitos de Temporización	11.9d
¿Están definidos los requisitos de temporización del software?	
Tamaño de memoria	11.9e
¿Se han descrito las limitaciones del tamaño de memoria?	
Interfaces hardware y software	11.9f
¿Se han descrito los interfaces hardware y software?	
¿Se han descrito los protocolos para los interfaces?	
Monitorización y detección de fallos	11.9g
¿Se han descrito los mecanismos de monitorización?	
¿Se han descrito los mecanismos de detección de fallos?	
Requisitos de particionamiento	11.9h

Tabla 14. Lista de comprobación del SRD

Objetivo	Referencia DO-178B
¿Se han descrito las particiones del software?	
¿Se han descrito los requisitos control entre las distintas particiones?	

2.3.4. Proceso de Diseño

Los objetivos del proceso de diseño comienzan con la mención a los requisitos de bajo nivel. Estos requisitos deben venir desarrollados desde los requisitos software de alto nivel, y suelen ser un refinamiento de los mismos, que se reflejan en el diseño.

En este punto, es necesario hacer una aclaración. El proceso de desarrollo software producirá uno o más niveles de requisitos software. Los requisitos software de alto nivel vienen desde los requisitos y arquitectura de sistema. Normalmente estos requisitos se desarrollan durante el proceso de diseño, creando uno o más niveles sucesivos de requisitos, que son los requisitos software de bajo nivel. No obstante, la DO-178B (Ref. [4]) habla de la posibilidad de que si el código fuente es desarrollado directamente a partir de los requisitos de alto nivel, entonces éstos son considerados también requisitos de bajo nivel, y las consideraciones para estos requisitos también son aplicables.

En la práctica, para algunas Autoridades de Certificación esto no siempre es una consideración aceptable, especialmente para niveles más altos del D.

Tabla 15. Lista de comprobación del proceso de Diseño

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Los requisitos software de bajo nivel cumplen con los requisitos software de alto nivel?	A-4, #1	6.3.2a	C
¿Los requisitos software de bajo nivel son precisos y consistentes?	A-4, #2	6.3.2b	C
¿Los requisitos software de bajo nivel son compatibles con el hardware?	A-4, #3	6.3.2c	B
¿Los requisitos software de bajo nivel son verificables?	A-4, #4	6.3.2d	B
¿Los requisitos software de bajo nivel cumplen con los estándares?	A-4, #5	6.3.2e	C
¿Los requisitos software de bajo nivel están trazados a los requisitos software de alto nivel?	A-4, #6	6.3.2f	C
¿Son precisos los algoritmos?	A-4, #7	6.3.2g	C
¿Es compatible la arquitectura software con los requisitos software de alto nivel?	A-4, #8	6.3.3a	C

Tabla 15. Lista de comprobación del proceso de Diseño

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Es consistente la arquitectura software?	A-4, #9	6.3.2b	C
¿Es compatible la arquitectura software con el hardware?	A-4, #10	6.3.3c	B
¿Es verificable la arquitectura software?	A-4, #11	6.3.3d	B
¿Cumple la arquitectura software con los estándares?	A-4, #12	6.3.3e	C
¿Está confirmada la integridad de las particiones software?	A-4, #13	6.3.3f	D

Dentro del proceso de diseño, se produce el SDD. Este documento puede contener los requisitos software de bajo nivel, así como la arquitectura software e información más detallada sobre el diseño software. Deben ser descritos los componentes software, los métodos de diseño y particionamiento, el flujo de control y el flujo de datos.

Tabla 16. Lista de comprobación del SDD

Objetivo	Referencia DO-178B
Cumplimiento con los requisitos de alto nivel	11.10a
¿Se ha descrito cómo el software satisface los requisitos de alto nivel?	
Descripción de la arquitectura software	11.10b
¿Se ha descrito la arquitectura software?	
Descripción de las entradas y salidas	11.10c
¿Se han descrito las entradas y salidas a través de la arquitectura software? Esto incluye, por ejemplo, los diccionarios de datos.	
Flujo de control y de datos	11.10d
¿Se ha descrito el flujo de control y de datos del diseño?	
Limitaciones en los recursos	11.10e
¿Se han descrito las limitaciones de los recursos? Por ejemplo la temporización y la memoria.	
Procesos de planificación de tareas	11.10f
¿Se han descrito los procesos de planificación de tareas?	
Métodos de diseño	11.10g
¿Se han descrito los métodos de diseño y detalles para su implementación?	

Tabla 16. Lista de comprobación del SDD

Objetivo	Referencia DO-178B
Métodos de particionamiento	11.10h
¿Se han definido los mecanismos de protección de las particiones?	
Descripción de los componentes software	11.10i
¿Se han descrito los componentes software? Si son previamente desarrollados, es necesario especificar información acerca de su origen.	
Requisitos derivados	11.10j
¿Se han definido y justificado los requisitos derivados del proceso de diseño software?	
Tratamiento del código desactivado	11.10k
Si el sistema contiene código desactivado, ¿existe una descripción del modo en que se asegura que este código no va a ser ejecutado en el sistema?	
Decisiones de diseño	11.10l
¿Existe una justificación de las decisiones de software relacionadas con los requisitos safety?	

2.3.5. Proceso de Implementación

El proceso de implementación se corresponde con la codificación e integración del código fuente desarrollado a partir de los requisitos de bajo nivel. Esta lista de comprobación sólo es aplicable a partir de un nivel C, donde se necesita comprobar que el código fuente está de acuerdo con la arquitectura software, los requisitos de bajo nivel, cumplimiento con los estándares. Además, para un nivel B, es necesario comprobar y evidenciar que el código fuente es verificable.

La lista de comprobación que se muestra a continuación contiene preguntas acerca de todos los objetivos que son necesarios cubrir según la DO-178B (Ref. [4]) en el proceso de implementación.

Tabla 17. Lista de comprobación del proceso de Implementación

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Está de acuerdo el código fuente con los requisitos software de bajo nivel?	A-5, #1	6.3.4a	C
¿Está de acuerdo el código fuente con la arquitectura software?	A-5, #2	6.3.4b	C
¿Es verificable el código fuente?	A-5, #3	6.3.4c	B

Tabla 17. Lista de comprobación del proceso de Implementación

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Cumple el código fuente con los estándares?	A-5, #4	6.3.4d	C
¿Está trazado el código fuente con los requisitos software de bajo nivel?	A-4, #5	6.3.4e	C
¿Es preciso y consistente el código fuente?	A-4, #6	6.3.4f	C
¿Es completa y correcta la salida del proceso de integración software?	A-5, #7	6.3.5	C

2.3.6. Proceso de Integración

El proceso de integración debe asegurar el cumplimiento con los requisitos de alto nivel y bajo nivel. También es necesario comprobar que el código objeto ejecutable es robusto con los requisitos, así como que es compatible con la plataforma hardware.

Tabla 18. Lista de comprobación del proceso de Integración

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Cumple el código objeto ejecutable con los requisitos software de alto nivel?	A-6, #1	6.4.2.1 6.4.3	D
¿Es robusto el código objeto ejecutable con los requisitos software de alto nivel?	A-6, #2	6.4.2.2 6.4.3	D
¿Cumple el código objeto ejecutable con los requisitos software de bajo nivel?	A-6, #3	6.4.2.1 6.4.3	C
¿Es robusto el código objeto ejecutable con los requisitos software de bajo nivel?	A-6, #4	6.4.2.2 6.4.3	C
¿Es compatible el código objeto ejecutable con el hardware?	A-6, #5	6.4.3a	D

2.3.7. Proceso de Verificación

También es necesario comprobar que el proceso de verificación se ha llevado a cabo correctamente. Para ello es necesario comprobar los procedimientos de pruebas, si los resultados han sido satisfactorios, y, dependiendo del nivel aplicable, el grado y tipo de cobertura estructural alcanzado. Además, es necesario asegurar que han sido definidas todas las pruebas al nivel necesario aplicable.

Tabla 19. Lista de comprobación del proceso de Verificación

Objetivo	Objetivo de la tabla	Referencia DO-178B	Nivel mínimo Aplicable
¿Son correctos los procedimientos de pruebas?	A-7, #1	6.3.6b	C
¿Son correctos los resultados de las pruebas y están explicadas las discrepancias?	A-7, #2	6.3.6c	C
¿Se ha conseguido la cobertura completa de las pruebas para los requisitos software de alto nivel?	A-7, #3	6.4.4.1	D
¿Se ha conseguido la cobertura completa de las pruebas para los requisitos software de bajo nivel?	A-7, #4	6.4.4.1	C
¿Se ha conseguido la cobertura estructural completa de condición/decisión modificada (MC/DC)?	A-7, #5	6.4.4.2	A
¿Se ha conseguido la cobertura estructural completa de decisiones?	A-7, #6	6.4.4.2a 6.4.4.2b	B
¿Se ha conseguido la cobertura estructural completa de decisiones?	A-7, #7	6.4.4.2a 6.4.4.2b	C
¿Se ha conseguido la cobertura estructural completa (flujo de control y flujo de datos)?	A-7, #8	6.4.4.2c	C

CAPÍTULO 3: AVEMACS: HERRAMIENTA PARA LA GESTIÓN DE LA VERIFICACIÓN SOFTWARE EN SISTEMAS CRÍTICOS

En este capítulo se presentará una especificación general de la herramienta, se describirá la interfaz de usuario, la funcionalidad y finalmente el entorno de desarrollo y ejecución.

3.1. ESPECIFICACIÓN GENERAL DE LA HERRAMIENTA

La herramienta tiene dos módulos bien diferenciados:

- Módulo de Administración: Proporciona utilidades para la administración de la herramienta.
- Módulo de Gestión de la verificación: Proporciona un medio para la gestión de los procesos de verificación dentro del ciclo de vida software.

En los siguientes sub-apartados se describe con más detalle cada uno de estos módulos.

3.1.1. Módulo de Administración

El módulo de administración tiene como misión principal la gestión de la herramienta. Desde este módulo se podrá:

- Gestionar usuarios
- Gestionar los permisos
- Gestionar las propiedades de los proyectos
- Gestionar las checklists de los procesos
- Gestionar las checklists de los documentos
- Gestionar los documentos

El siguiente diagrama muestra todas las operaciones y sub-operaciones disponibles dentro del módulo de administración:

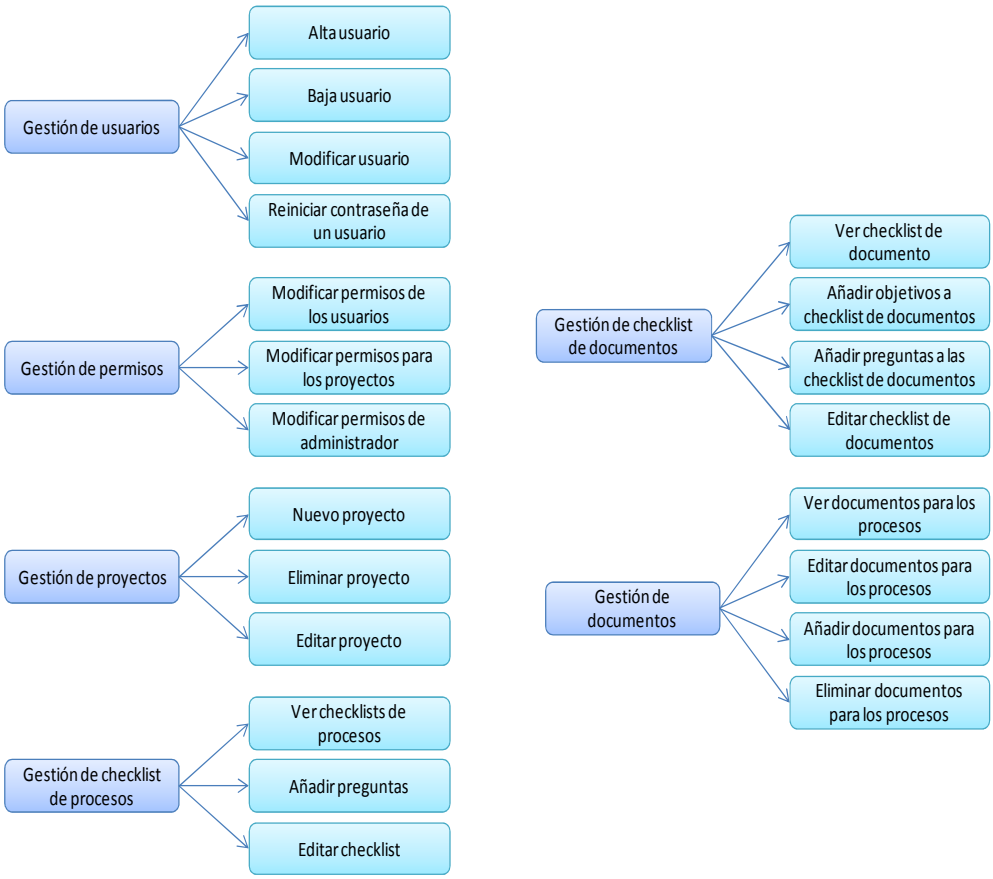


Figura 7. Módulo de administración – Operaciones permitidas

La pantalla de administración mostrará todos los grupos de operaciones de administración. El flujo de control para este proceso es muy sencillo:

Figura 8. Módulo de administración – Pantalla principal. Flujo de control

Debido a que todas las operaciones de administración estarán disponibles para todos los administradores de la herramienta, habrá una función que simplemente muestre las operaciones por pantalla, en una lista.

A continuación se presentan las operaciones del módulo de administración.

3.1.1.1. Gestión de usuarios

Las operaciones que se pueden realizar son las siguientes:

- Alta de nuevos usuarios.
- Modificación de los datos de los usuarios
- Baja de usuarios
- Reinicio de la contraseña de un usuario

La pantalla de operaciones para la gestión de usuarios tiene un flujo de control que incluye únicamente la presentación de las operaciones disponibles:

Figura 9. Módulo de administración – Gestión de usuarios. Flujo de control

3.1.1.1.1. Alta de nuevos usuarios

Cuando se desea dar el alta de un nuevo usuario en la herramienta, se debe definir una serie de información obligatoria para cada usuario. Esta información es la siguiente:

- Nombre y apellidos del usuario: Con esta información se puede identificar al usuario.

- Nombre del usuario: Es el nombre de usuario que se utilizará para acceder a la herramienta. Se recomienda que el nombre de usuario sea el mismo que el utilizado en los sistemas corporativos, para una identificación inequívoca.
- Iniciales del usuario: Estas iniciales sirven para tener una identificación abreviada en los informes de verificación y en las incidencias que se registren en las herramientas.
- Dirección de e-mail: Se utiliza para tener en la herramienta la información necesaria para el contacto con el usuario.

Cuando se da de alta a un usuario, se establece una contraseña por defecto, que el usuario puede cambiar en cualquier momento. La contraseña por defecto es igual al nombre de usuario.

El flujo de control del alta de nuevos usuarios está descrito en la Figura 10.

Cuando se validan los datos del formulario, se comprueba lo siguiente:

- Se han introducido todos los datos obligatorios
- El correo electrónico introducido tiene el formato correcto.
- El nombre de usuario introducido no está registrado previamente en la base de datos.

Una vez se han realizado todas las comprobaciones, se almacenan los datos en la base de datos.

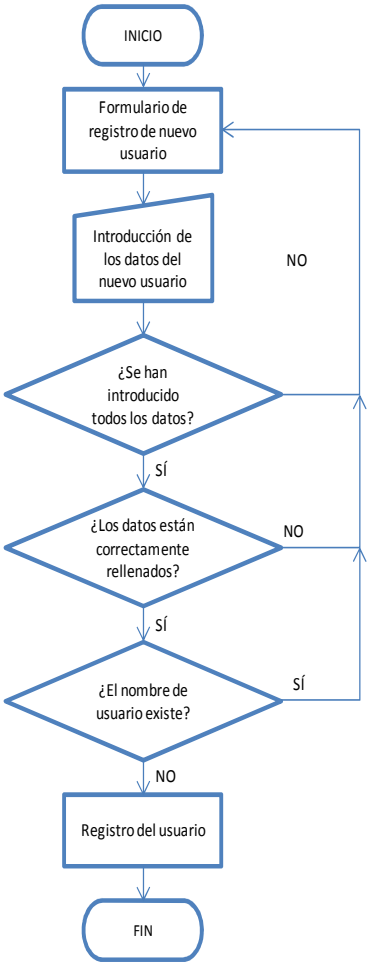


Figura 10. Módulo de administración – Alta de un nuevo usuario. Flujo de control

3.1.1.1.2. Modificación de los datos de los usuarios

Es posible modificar todos los datos que se definieron al crear un usuario. Para ello, una vez seleccionado el usuario, aparecerán los datos previamente almacenados y será posible modificarlos. Antes de modificar los datos de un usuario, se realizarán las mismas comprobaciones que en el caso del alta de nuevo usuario.

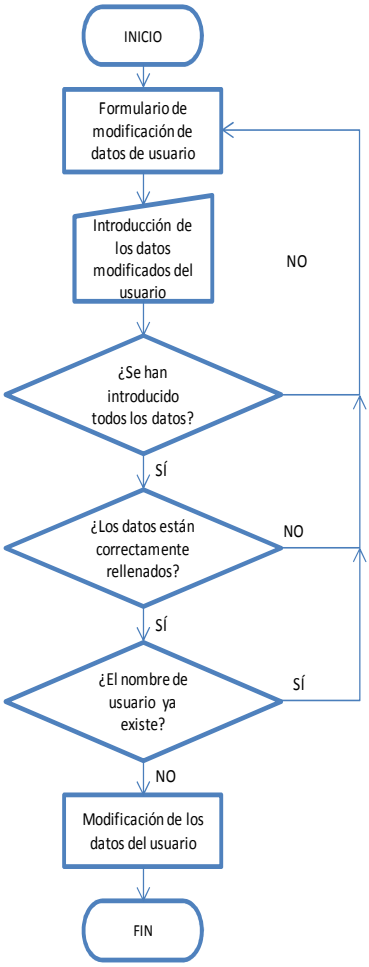


Figura 11. Módulo de administración – Modificación de datos de usuario. Flujo de control

3.1.1.1.3. Baja de usuarios

Es posible dar de baja de la base de datos a todos los usuarios excepto a los administradores. Una vez seleccionado el usuario, se eliminarán sus datos de la base de datos.

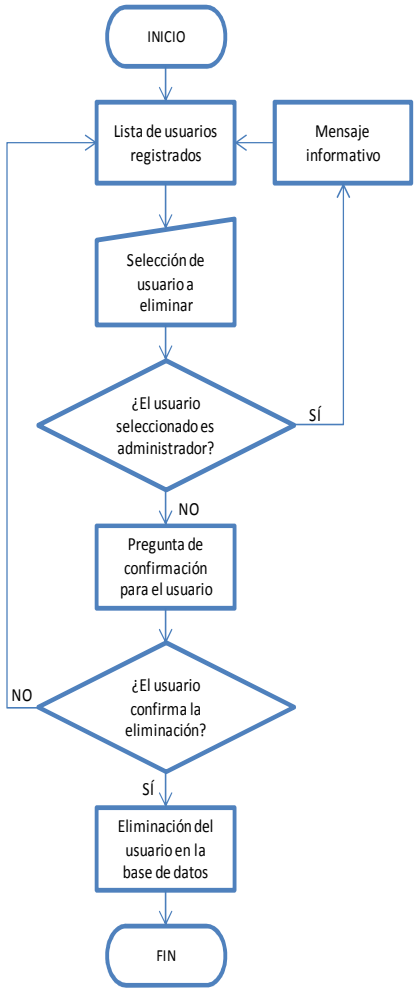


Figura 12. Módulo de Administración – Baja de un usuario. Flujo de control

3.1.1.1.4. Reinicio de la contraseña de un usuario

Es posible reiniciar la contraseña de cualquier usuario excepto de los administradores de la herramienta. Para ello, después de seleccionar el usuario a reiniciar la contraseña, se pedirá una confirmación por parte del usuario.

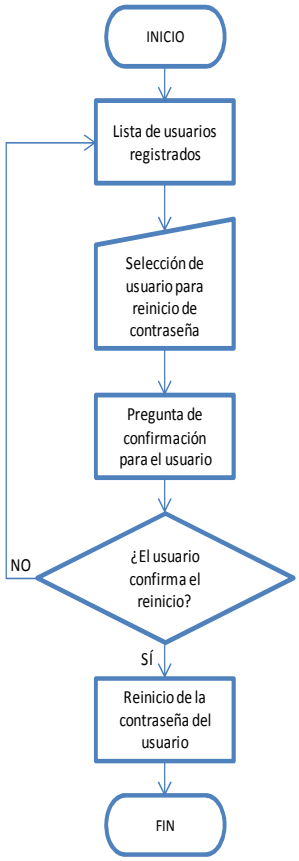


Figura 13. Módulo de Administración – Reinicio de contraseña. Flujo de control

3.1.1.2. Gestión de permisos

La gestión de permisos se realiza a tres niveles: gestión de los permisos de los usuarios para los proyectos, gestión de los permisos de un proyecto y gestión de los permisos de administración de la herramienta.

Figura 14. Módulo de administración – Gestión de permisos. Flujo de control

Es posible seleccionar un usuario para gestionar sus permisos dentro de los distintos proyectos. Una vez seleccionado el usuario, aparecerá una tabla con todos los proyectos registrados en la herramienta y los distintos niveles de permisos posibles. Se podrá seleccionar el nivel de acceso independientemente para cada uno de los proyectos.

Cuando se quieren modificar todos los permisos de un proyecto concreto, aparecerán todos los usuarios registrados en la herramienta, y es posible seleccionar individualmente el nivel de acceso para cada uno de los usuarios.

El nivel de acceso de administración permite realizar las operaciones de administración de la herramienta, además de tener acceso a todas las operaciones disponibles para la gestión de la verificación en todos los proyectos. Es posible seleccionar cualquier usuario para ser administrador, pero no es posible eliminar los permisos de administración de un usuario "maestro", para evitar que se eliminen los permisos de administración para todos los usuarios.

3.1.1.3. Gestión de los proyectos

Esta funcionalidad incluye tres operaciones: creación de un nuevo proyecto, eliminación de un proyecto y edición de las propiedades del proyecto.

Cuando se crea un nuevo proyecto, es necesario introducir unas propiedades básicas, como son el nombre del proyecto, la normativa aplicable y el nivel de aseguramiento necesario para el proyecto. En caso de que no se hayan introducido todos los datos, no se podrá registrar el proyecto.

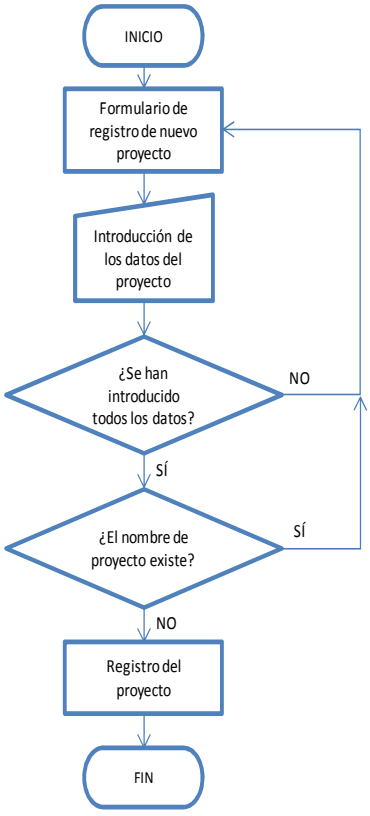


Figura 15. Módulo de Administración. Registro de nuevo proyecto. Flujo de control

Se puede eliminar un proyecto de la base de datos, simplemente seleccionando la operación desde el menú de administración. Antes de eliminar el proyecto, se pedirá una confirmación por parte del usuario.

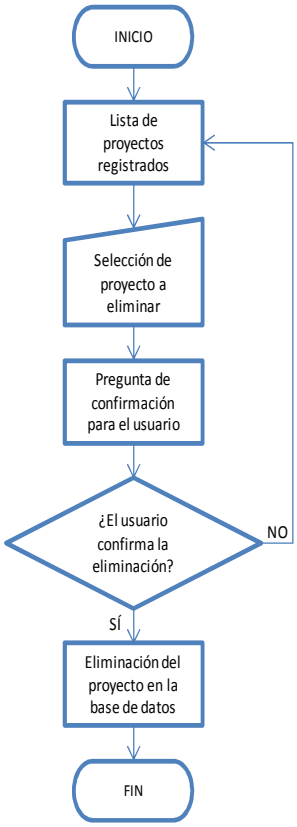


Figura 16. Módulo de Administración. Eliminación de un proyecto. Flujo de control

Una vez que se ha creado el proyecto, es posible editar las propiedades que se definieron a la hora de crear el mismo.

3.1.1.4. Gestión de las listas de comprobación de los procesos

Las listas de comprobación son una forma rápida de conocer el estado de verificación de un proceso. Estas listas de comprobación están orientadas a los objetivos de la normativa aplicable para el proyecto. Los distintos objetivos para cada normativa dependen del nivel de aseguramiento necesario.

La herramienta permite definir las preguntas para las distintas listas de comprobación de los procesos dentro de una normativa concreta. Para ello, se seleccionará la normativa aplicable y el proceso dentro de dicha normativa.

Los datos obligatorios para la pregunta son: el texto de la pregunta, el objetivo de la tabla, la referencia a la normativa y el nivel mínimo aplicable para cada pregunta.

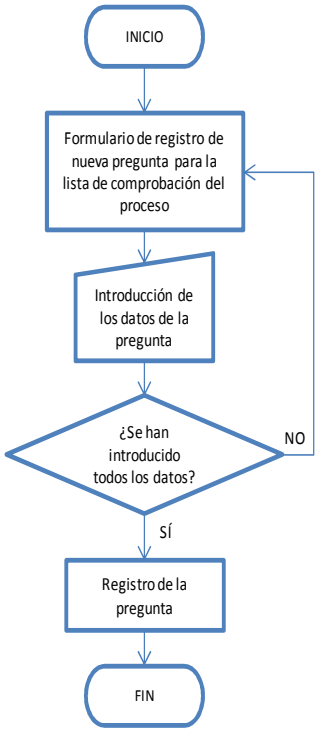


Figura 17. Módulo de Administración. Gestión de listas de comprobación para los procesos. Añadir una nueva pregunta

Una vez creadas las preguntas, es posible editar esta lista de comprobación para cada proceso específico. Es posible tanto editar las preguntas definidas como eliminar las mismas de la lista de comprobación.

Cuando se va a editar una pregunta, es necesario que queden rellenados todos los campos obligatorios que se definieron cuando se creó la pregunta.

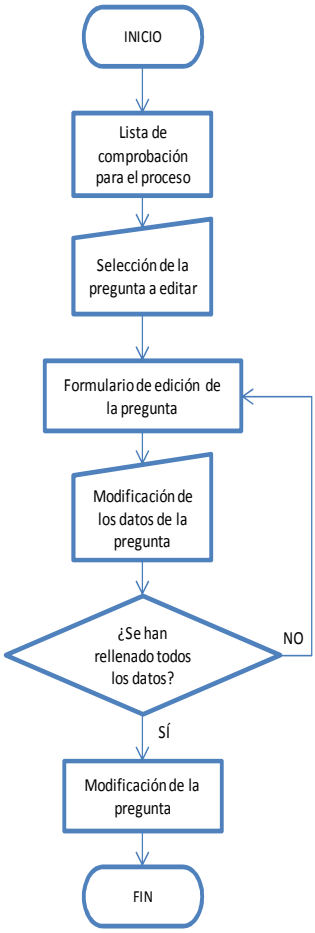


Figura 18. Módulo de Administración. Gestión de listas de comprobación para los procesos. Editar una pregunta

También se puede consultar el contenido de cada una de las listas de comprobación. Para ello, se seleccionará la normativa aplicable y el proceso dentro de dicha normativa.

3.1.1.5. Gestión de las listas de comprobación de los documentos

De la misma forma que se definen las listas de comprobación de los procesos, es posible definir las listas de comprobación de los documentos. Estas listas están estructuradas de forma distinta a las de procesos, ya que es posible definir tanto los objetivos de la normativa como las preguntas asociadas a esos objetivos.

Las operaciones de gestión de las listas de comprobación de los documentos son la definir objetivos, definir preguntas para los objetivos, edición de las listas de comprobación y visualización de las mismas.

Para añadir un objetivo a una lista de comprobación de un documento, es obligatorio especificar el texto del objetivo y la referencia a la normativa aplicable.

Para añadir una pregunta es necesario incluir el texto de la pregunta y el objetivo al cual es aplicable la misma.

Cuando se va a editar una pregunta o un objetivo, es obligatorio introducir todos los campos que son necesarios cuando se añaden los mismos.

Cuando se elimine un objetivo, se eliminarán las preguntas asociadas a dicho objetivo. De esta forma no quedarán preguntas sin tener un objetivo asociado.

3.1.1.6. Gestión de los documentos

Es posible definir los documentos o elementos de configuración para los distintos procesos, de tal forma que cuando se gestiona el proceso de verificación sea posible seleccionar los elementos de configuración aplicables a cada proyecto. Estos elementos de configuración son principalmente documentos, aunque se pueden definir también otro tipo de elementos, como puede ser el código fuente, que sería aplicable al proceso de verificación de la implementación.

Es posible añadir, eliminar, editar o visualizar elementos de configuración. La información necesaria para crear un elemento de configuración es la normativa y el proceso al que aplica, el nombre del documento, el acrónimo y el nivel mínimo de aplicación.

3.1.2. Módulo de Gestión de la Verificación

El módulo de gestión de la verificación tiene como principal aplicación la gestión de los procesos de verificación dentro del ciclo de vida software de los proyectos.

Dentro del módulo de gestión de la verificación se pueden realizar las siguientes operaciones:

- Selección de proyecto
- Selección del proceso del ciclo de vida
- Ver/Editar lista de comprobación del proceso
- Ver/Editar lista de comprobación de un documento
- Ver/Editar incidencias de un documento
- Añadir un documento a un proceso
- Editar datos de un documento
- Eliminar un documento
- Ver propiedades de los usuarios del proyecto

El siguiente diagrama muestra las operaciones que se pueden realizar dentro del módulo de Gestión de la Verificación.

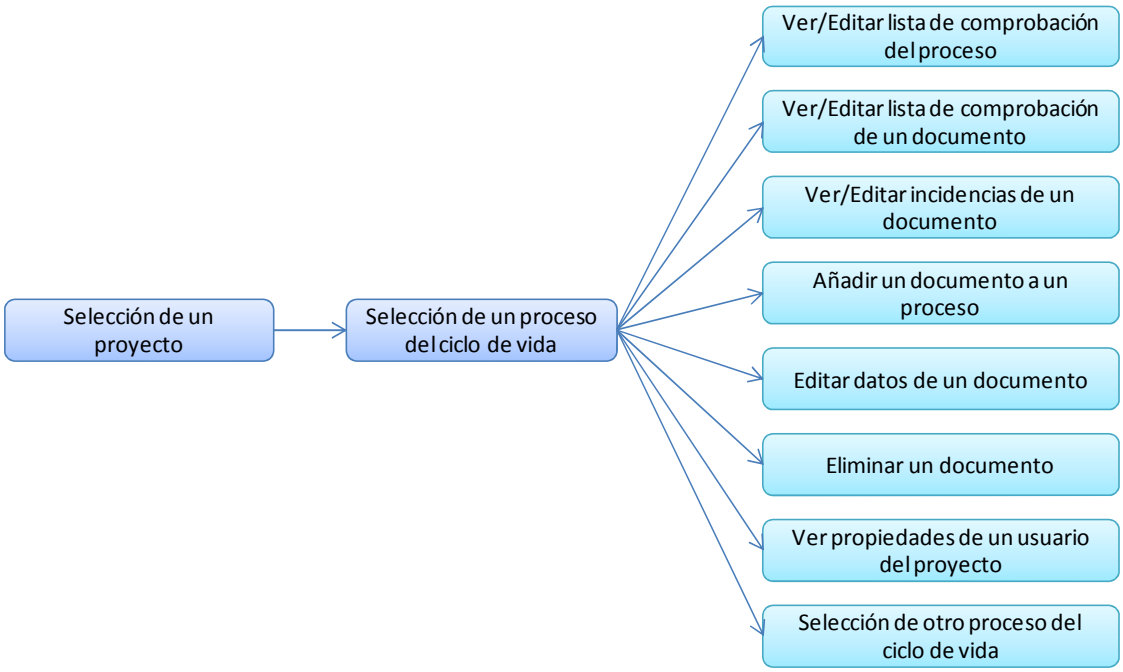


Figura 19. Módulo de Gestión de la Verificación – Operaciones permitidas

Los permisos de lectura y escritura en las operaciones dependerán del rol del usuario dentro del proyecto.

Se han definido los siguientes procesos dentro del ciclo de vida del software:

- Proceso de planificación
- Proceso de desarrollo
- Proceso de requisitos
- Proceso de diseño
- Proceso de implementación
- Proceso de integración
- Proceso de verificación

Cuando se selecciona un proyecto, es posible conocer las propiedades del mismo, así como el estado general del proceso de verificación y los usuarios que forman parte del proyecto, junto con el rol para cada usuario.

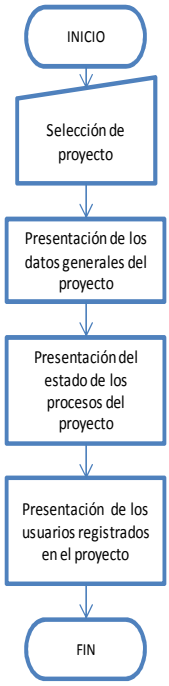


Figura 20. Módulo de Gestión de la Verificación. Presentación general del proyecto. Flujo de control

La información general del proyecto consta del nombre del proyecto, la normativa aplicable y el nivel de aseguramiento del proyecto.

El estado de la verificación de cada uno de los procesos puede tener dos estados:

- Pendiente: Este estado se presenta cuando se da una de las siguientes condiciones:
  - La lista de comprobación para el proceso tiene registrada al menos una respuesta negativa.
  - La lista de comprobación para cualquiera de los documentos del proceso tiene registrada al menos una respuesta negativa.
  - Hay abierta alguna incidencia (comentario) acerca de alguno de los documentos.
- Correcto: Para que el estado de la verificación del proceso esté completo y correcto, no todas las preguntas de las listas de comprobación deben ser afirmativas y los documentos no deben tener ningún comentario o incidencia abiertos.

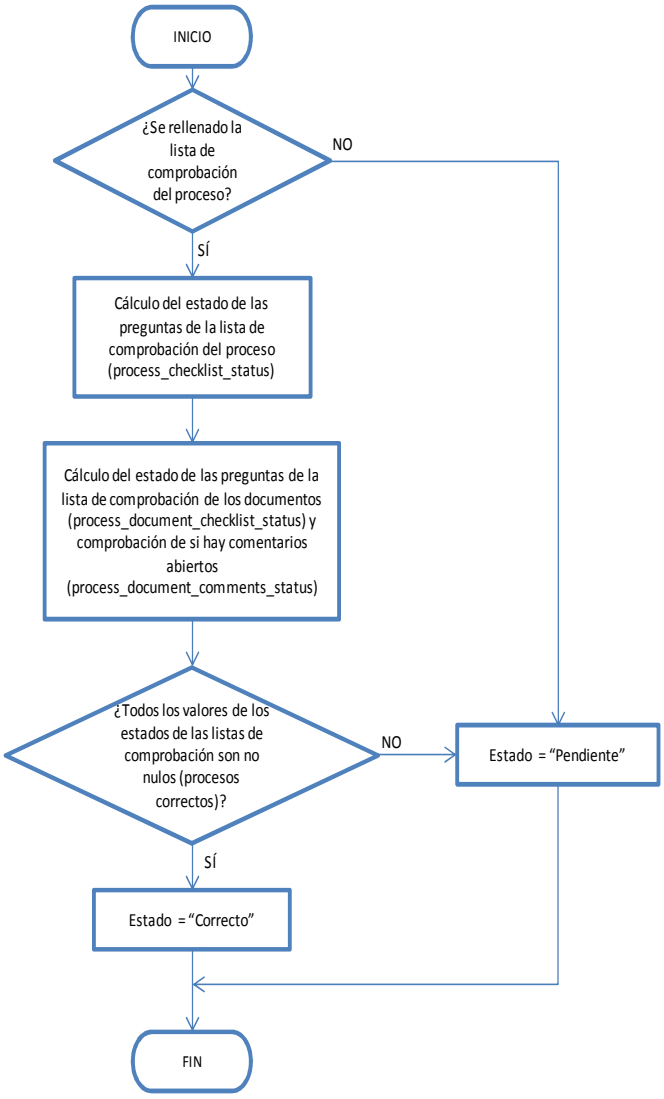


Figura 21. Módulo de Gestión de la Verificación. Presentación del estado de los procesos. Flujo de control

El estado de los procesos se identificará mediante un icono.

Para la presentación de los usuarios, simplemente se extraerán los usuarios de la base de datos, diferenciándolos según el rol con el que estén registrados en el proyecto.

Para cada uno de los procesos definidos, existe una lista de comprobación donde será posible responder a las distintas preguntas definidas (si el usuario tiene los permisos necesarios), así como introducir comentarios adicionales para aclarar o detallar la respuesta a cada pregunta.

Dentro de cada proceso, es posible añadir documentos/elementos de configuración para su seguimiento. Para ello, es necesario especificar como mínimo el nombre y el tipo de documento. Si se conoce, es posible añadir el código del documento. En el caso de que el código del documento no esté disponible, la herramienta automáticamente presentará el estado del código como "Pendiente".

Para cada uno de los documentos registrados para el proyecto, estará disponible una lista de comprobación y una lista de incidencias. La lista de comprobación presentará las preguntas y, con los permisos de usuario necesarios, será posible responder a las preguntas e introducir comentarios a las respuestas.

Cuando se detecta una no conformidad con un documento específico, se introducirá un comentario para ese documento.

Dependiendo del proceso al que pertenezca el documento, existen una serie de campos para los comentarios. Para el caso del proceso de planificación, se indica la edición del documento, el comentario de verificación, la respuesta por parte del desarrollo, la versión en la que se da por cerrado el comentario y el estado.

En los procesos en los que sea susceptible de necesidad la utilización de un estándar (caso de los procesos de requisitos, diseño e implementación), además se puede introducir la regla del estándar a la que se refiere el comentario. Dentro de este grupo de procesos, en el caso del proceso de requisitos y de diseño, hay posibilidad de indicar el requisito del que se especifica el comentario. En el caso del proceso de implementación, aparecen los campos de fichero y línea de código.

En todos los casos se queda registrado el usuario que realiza el comentario.

3.1.3. Diseño de la Base de Datos

La herramienta contiene una base de datos con las siguientes tablas:

- t_admin_users
- t_assurance_levels
- t_documents
- t_document_answers
- t_document_checklists
- t_document_comments
- t_document_objectives
- t_document_questions
- t_normatives
- t_processes
- t_process_answers
- t_process_questions
- t_projects
- t_project_documents
- t_users
- t_user_access

La descripción de cada una de las tablas se encuentra en los siguientes sub-apartados.

3.1.3.1. t_admin_users

Esta tabla contiene los usuarios administradores de la herramienta completa.

Los campos de la tabla son los siguientes:

Tabla 20. t_admin_users

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
admin_user_id	Int(11)	Ninguno	Auto incremento	Primario	ID de Administrador del usuario
user_id	Int(11)	Ninguno	-	user_id (t_users)	ID del usuario dentro de la herramienta.

3.1.3.2. t_assurance_levels

Esta tabla contiene los nombres de todos los niveles definidos dentro de cada normativa.
Los campos de la tabla son los siguientes:

Tabla 21. t_assurance_levels

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
level_id	Int(11)	Ninguno	Auto incremento	Primario	ID del nivel de aseguramiento
level_name	Varchar(6)	Ninguno	-		Nombre del nivel de aseguramiento
normative_id	Int(11)	Ninguno	-	normative_id (t_normatives)	ID de la normativa aplicable para el nivel

3.1.3.3. t_documents

Esta tabla contiene todos los documentos genéricos definidos dentro de la normativa.
Los campos de la tabla son los siguientes:

Tabla 22. t_documents

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_id	Int(11)	Ninguno	Auto incremento	Primario	ID del documento
document_name	Varchar(128)	NULL	-	-	Nombre del documento
document_acr	Varchar(10)	NULL	-	-	Acrónimo para el documento
normative_id	Int(11)	Ninguno	-	normative_id (t_normatives)	ID de la normativa aplicable para el documento
process_id	Int(11)	Ninguno	-	process_id (t_processes)	ID del proceso aplicable para el documento

Tabla 22. t_documents

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
level_id	Int(11)	Ninguno	-	level_id (t_assurance_levels)	ID del nivel mínimo en el que el documento es aplicable

3.1.3.4. t_document_answers

Esta tabla contiene las respuestas y el estado de las mismas para las preguntas de las listas de comprobación para un documento de un proyecto concreto.

Los campos de la tabla son los siguientes:

Tabla 23. t_document_answers

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_answer_id	Int(11)	Ninguno	Auto incremento	Primario	ID de la respuesta
project_document_id	Int(11)	Ninguno	-	project_document_id (t_project_documents)	ID del documento al que aplica la respuesta
document_question_id	Int(11)	Ninguno	-	document_question_id (t_document_questions)	ID de la pregunta de la lista de comprobación a la que aplica la respuesta
document_answer_text	Varchar(128)	NULL	-	-	Texto de la respuesta
document_answer_result	Tinyint(1)	0	-	-	Resultado de la respuesta (0-NO, 1-SÍ)

3.1.3.5. t_document_checklists

Esta tabla relaciona cada documento genérico con un ID de lista de comprobación, para poder asociar las preguntas.

Los campos de esta tabla son los siguientes:

Tabla 24. t_document_checkists

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_checklist_id	Int(11)	Ninguno	Auto incremento	Primario	ID de la lista de comprobación del documento genérico
document_id	Int(11)	Ninguno	-	document_id (t_documents)	ID del documento al que aplica la lista de comprobación

3.1.3.6. t_document_comments

Esta tabla contiene todos los comentarios asociados a los documentos específicos para cada proyecto.

Los campos de esta tabla son los siguientes:

Tabla 25. t_document_comments

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_comment_id	Int(11)	Ninguno	Auto Incremento	Primario	ID único del comentario para el documento de proyecto
project_document_id	Int(11)	Ninguno		Project_document_id (t_project_documents)	ID del proyecto al que aplica el comentario
documment_comment_author	Int(11)	NULL	-	User_id (t_users)	ID del autor del comentario
document_comment_edition_open	Varchar(8)	NULL	-	-	Edición en la que se detectó el comentario
document_comment_rule	Varchar(10)	NULL	-	-	Regla que inclumple el elemento de configuración
document_comment_requirement_id	Varchar(32)	NULL	-	-	ID del requisito para el comentario

Tabla 25. t_document_comments

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_comment_location	Varchar(8)	NULL	-	-	Lugar (línea de código o página) donde se detecta el comentario
document_comment_text	Varchar(512)	Ninguno	-	-	Texto del comentario
document_comment_response	Varchar(512)	NULL	-	-	Respuesta del comentario
document_comment_edition_close	Varchar(8)	NULL	-	-	Edición o versión del documento donde se da por cerrado del comentario
document_comment_status	Tinyint(1)	0	-	-	Estado del comentario (0-Abierto, 1-Cerrado)

3.1.3.7. t_document_objectives

Esta tabla contiene los objetivos para los documentos genéricos, así como la información sobre los mismos.

Los campos de la tabla son los siguientes:

Tabla 26. t_document_objectives

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_objective_id	Int(11)	Ninguno	Auto Incremento	Primario	ID del objetivo
document_checklist_id	Int(11)	Ninguno	-	Document_checklist_id (t_document_checklists)	ID de la checklist para el objetivo
document_objective_text	Varchar(256)	NULL	-		Texto del objetivo
document_objective_normative_objective	Varchar(16)	NULL	-		Referencia a la normativa

3.1.3.8. t_document_questions

Esta tabla contiene las preguntas de las listas de comprobación para los documentos.
Los campos de la tabla son los siguientes:

Tabla 27. t_document_questions

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_question_id	Int(11)	Ninguno	Auto Incremento	Primario	ID de la pregunta
document_objective_id	Int(11)	Ninguno	-	Document_objective_id (t_document_objectives)	ID del objetivo
document_question_text	Varchar(512)	NULL	-	-	Texto de la pregunta

3.1.3.9. t_normatives

Esta tabla contiene las normativas implementadas en la herramienta.
Los campos de la tabla son los siguientes:

Tabla 28. t_normatives

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
normative_id	Int(11)	Ninguno	Auto incremento	Primario	ID de la normativa
normative_name	Varchar(16)	Ninguno	-	-	Nombre de la normativa

3.1.3.10. t_processes

Esta tabla contiene los procesos definidos para cada normativa.
Los campos de la tabla son los siguientes:

Tabla 29. t_processes

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
process_id	Int(11)	Ninguno	Auto incremento	Primario	ID del proceso
process_name	Varchar(16)	Ninguno	-	-	Nombre del proceso
normative_id	Int(11)	Ninguno	-	Normative_id (t_normatives)	ID de la normativa a la que aplica el proceso

3.1.3.11. t_process_answers

Esta tabla contiene las respuestas para cada proceso de un proyecto concreto para las preguntas definidas en la lista de comprobación de los procesos.

Los campos de la tabla son las siguientes:

Tabla 30. t_process_answers

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
process_answer_id	Int(11)	Ninguno	Auto incremento	Primario	ID de la respuesta
project_id	Int(11)	Ninguno	-	-	ID de proyecto
process_question_id	Int(11)	Ninguno	-	-	ID de la pregunta de la lista de comprobación
process_answer_text	Varchar(128)	NULL	-	-	Texto de la respuesta
process_answer_result	Tinyint(1)	0	-	-	Resultado de la respuesta de la lista de comprobación (0-No, 1-Sí)

3.1.3.12. t_process_questions

Esta tabla contiene las preguntas de las listas de comprobación para los procesos.

Los campos de la tabla son los siguientes:

Tabla 31. t_process_questions

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
process_question_id	Int(11)	Ninguno	Auto Increment o	Primario	ID de la pregunta para el proceso
process_question_text	Varchar(256)	Ninguno	-	-	Texto de la pregunta
process_question_table_objective	Varchar(16)	NULL	-	-	Objetivo de las tablas de la normativa
process_question_normative_objetctive	Varchar(16)	NULL	-	-	Objetivo que cubre la pregunta
normative_id	Int(11)	Ninguno	-	normative_id (t_normatives)	ID de la normativa

Tabla 31. t_process_questions

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
process_id	Int(11)	Ninguno	-	process_id (t_processes)	ID del proceso
level_id	Int(11)	Ninguno	-	level_id (t_assurance_levels)	ID del nivel mínimo aplicable

3.1.3.13. t_projects

Esta tabla contiene la lista de proyectos registrados en la herramienta, así como la información básica de los mismos.

Los campos de la tabla son los siguientes:

Tabla 32. t_projects

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
project_id	Int(11)	Ninguno	Auto incremento	Primario	ID del proyecto
project_name	Varchar(128)	Ninguno	-	-	Nombre del proyecto
normative_id	Int(11)	Ninguno	-	normative_id (t_normatives)	ID de la normativa
level_id	Int(11)	Ninguno	-	level_id (t_assurance_levels)	ID del nivel para el proyecto

3.1.3.14. t_project_documents

Esta tabla contiene la lista de documentos registrados para un proyecto concreto.

Los campos de la tabla son los siguientes:

Tabla 33. t_project_documents

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
project_document_id	Int(11)	Ninguno	Auto Incremento	Primario	ID del documento de proyecto
project_document_name	Varchar(128)	Ninguno	-	-	Nombre del documento de proyecto
project_document_code	Varchar(14)	NULL	-	-	Código del documento del proyecto

Tabla 33. t_project_documents

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
document_id	Int(11)	Ninguno	-	document_id (t_documents)	ID del documento genérico
project_id	Int(11)	Ninguno	-	project_id (t_projects)	ID del proyecto al que pertenece el documento

3.1.3.15. t_users

Esta tabla contiene la lista de usuarios registrados en la herramienta.
Los campos de la tabla son los siguientes:

Tabla 34. t_users

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
user_id	Int(11)	Ninguno	Auto incremento	Primario	ID del usuario
username	Varchar(16)	Ninguno	-	-	Nombre de usuario
password	Char(32)	Ninguno	-	-	Contraseña (almacenada con MD5)
username_complete	Varchar(128)	TBD	-	-	Nombre completo del usuario
user_initials	Varchar(5)	Ninguno	-	-	Iniciales del usuario
user_email	Varchar(128)	Ninguno	-	-	Correo electrónico del usuario

3.1.3.16. t_user_access

Esta tabla contiene la lista de permisos para cada usuario en los proyectos.
Los campos de la tabla son los siguientes:

Tabla 35. t_users_access

Columna	Tipo	Predeterminado	Extra	Índice	Descripción
access_id	Int(11)	Ninguno	Auto Incremento	Primario	ID del acceso
user_id	Int(11)	Ninguno	-	user_id (t_users)	ID del usuario
project_id	Int(11)	Ninguno	-	project_id (t_projects)	ID del proyecto
access_level	Int(11)	Ninguno	-	-	Nivel de acceso

3.2. INTERFAZ DE USUARIO Y FUNCIONALIDAD

Esta sección describe el interfaz de usuario a modo de manual de usuario.

3.2.1. Acceso a la herramienta

Para acceder a la herramienta, es necesario abrir un navegador e introducir la dirección para acceder a la herramienta. En el caso del prototipo, está instalada localmente, con lo que la dirección es http://localhost/avemacs/index.php.

Aparecerá una pantalla que solicita nombre de usuario y contraseña:

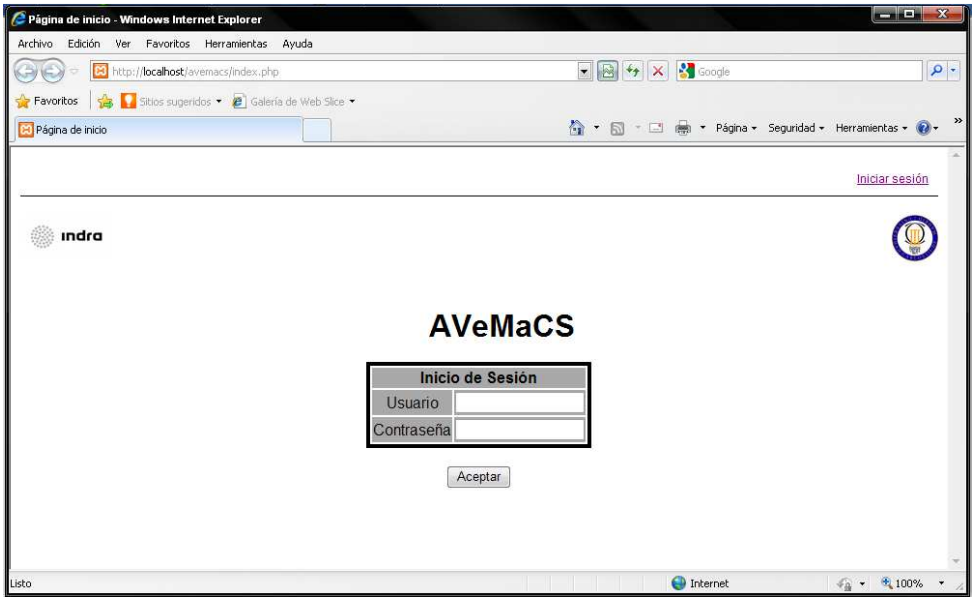


Figura 22. Acceso a la herramienta

Tras la introducción correcta del nombre de usuario y la contraseña correspondiente, se accederá a la página principal.

3.2.2. Página principal

Una vez se ha introducido un usuario y una contraseña válidos, se accede a la página principal de la herramienta. La pantalla principal mostrará todos los proyectos a los que pertenece el usuario.

En el caso de que el usuario que ha accedido sea administrador, aparecerán todos los proyectos, independientemente de si tiene acceso o no. En el caso de que el usuario pertenezca a un proyecto, al lado del mismo aparecerá indicándolo una estrella.

Para acceder a las propiedades del proyecto, basta con pulsar encima del nombre del mismo.

Si el usuario es administrador, en la página principal aparecerá una pestaña desde la que tendrá acceso a todas las opciones de administración de la herramienta.

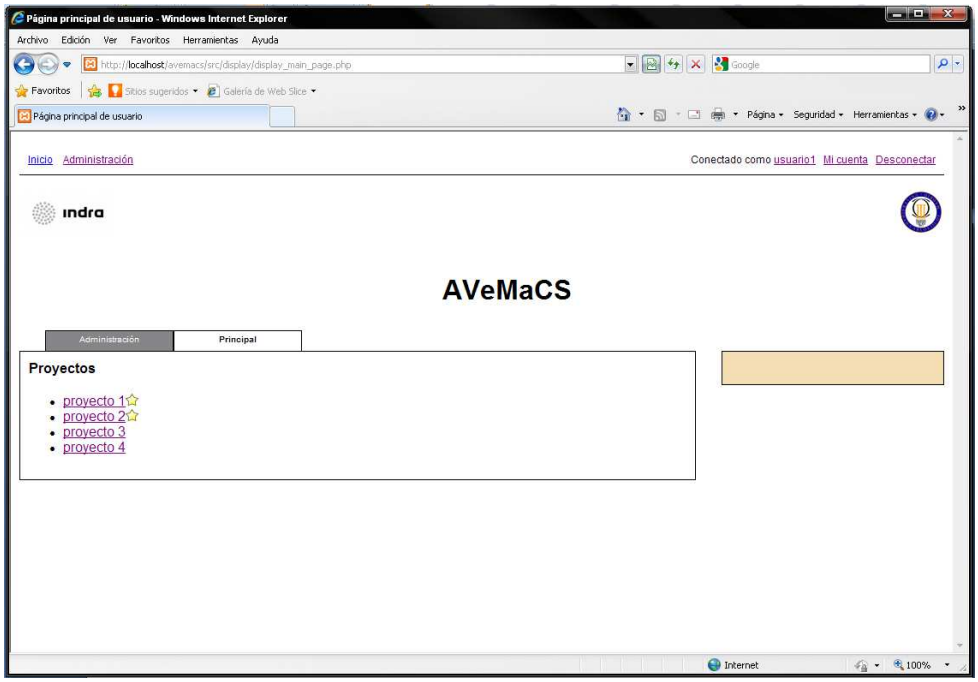


Figura 23. Página principal del usuario

3.2.3. Página de administración

Desde el menú de administración se pueden seleccionar todas las operaciones para gestión de la herramienta:

- Gestión de usuarios
- Gestión de permisos
- Gestión de proyectos
- Gestión de checklists de procesos
- Gestión de checklists de documentos
- Gestión de documentos

3.2.4. Página de estado y propiedades de un proyecto

Esta página está estructurada en tres partes:

- Información general del proyecto
- Estado del proyecto
- Usuarios pertenecientes al proyecto

Es aspecto de la pantalla se puede ver en la Figura 24.

Los datos que se presentan para el proyecto son datos básicos: nombre, normativa y nivel aplicable.

Junto al nombre de cada uno de los procesos, aparece una imagen, que podrá ser una de las dos siguientes:

- : Si se han introducido datos para el proceso y no existen incidencias relacionadas con el mismo.
- : Si no se han introducido datos para el proceso o el proceso tiene incidencias abiertas.

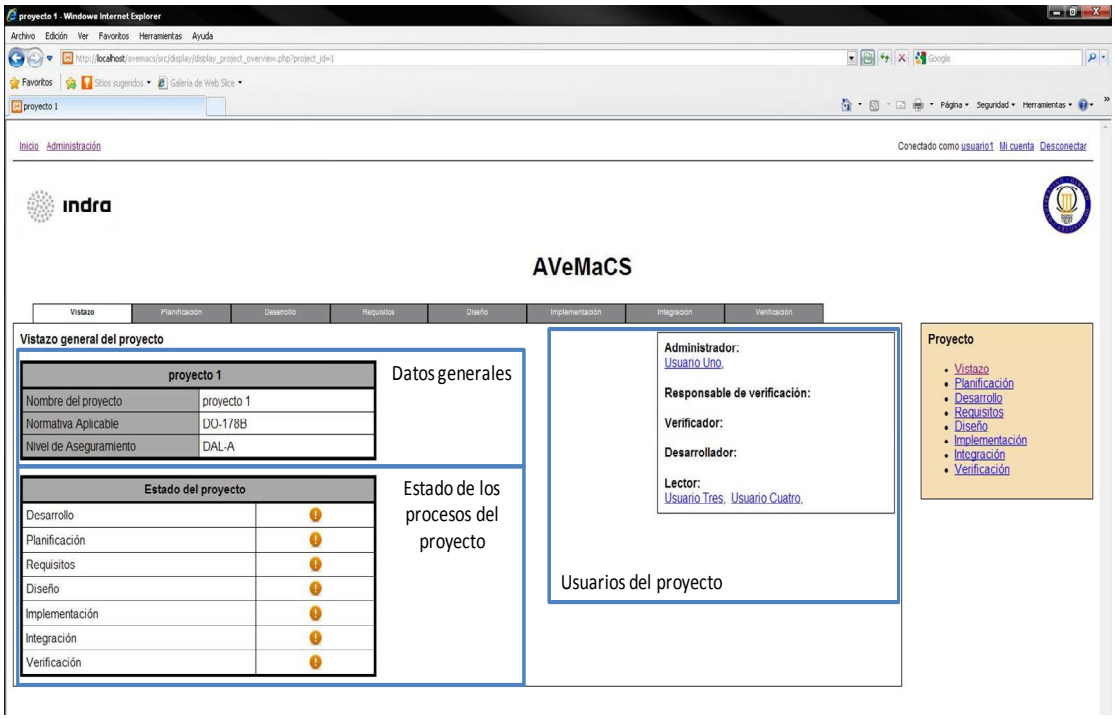


Figura 24. Página de estado y propiedades de un proyecto

Las incidencias para un proceso pueden aparecer en la listas de comprobación para el proceso, para las listas de comprobación de cualquiera de los documentos que se hayan registrado para el proceso o para los mismos documentos.

Cuando se hayan corregido todas las incidencias para un proceso, el icono pasará automáticamente a mostrar que no existen incidencias abiertas para el mismo.

El icono de estado de cada uno de los proyectos es también un hiperenlace a la página de información del proceso para el proyecto.

La lista de usuarios para el documento presenta los usuarios de acuerdo al rol dentro de el proyecto actual. Se puede conocer las propiedades un usuario pinchando con el ratón en el nombre del mismo.

Además de poder saltar por los distintos procesos a través de las pestañas, es posible utilizar los enlaces que aparecen en la parte derecha de la pantalla.

3.2.5. Página de información de un proceso del proyecto

Cuando se pulsa sobre cualquiera de los procesos, se pasa a la página de información de un proceso del proyecto.

La página de información de cada proceso está dividida en las siguientes partes:

- Estado de la lista de comprobación del proceso

- Información y estado de los documentos de un proceso
- Acciones disponibles para el proceso

El estado de la lista de comprobación de un proceso indica si el resultado de la misma tiene alguna pregunta cuya respuesta es negativa o por el contrario la respuesta a todas las preguntas tiene un resultado correcto. Este estado se indica mediante los siguientes iconos:

- : Si se han introducido los datos para todas las respuestas y el resultado es positivo para todas ellas.
- : Si no se han introducido datos para la lista de comprobación o existen preguntas con una respuesta negativa.

Para acceder al contenido de la lista de comprobación hay que pulsar sobre el icono de la lista.

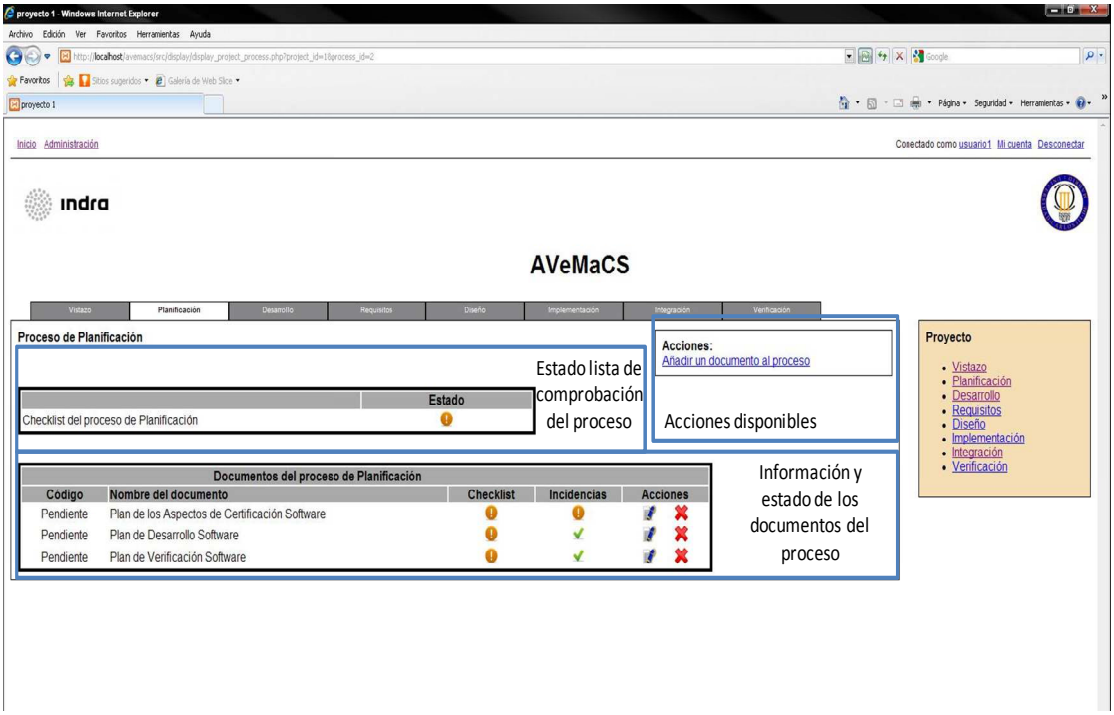


Figura 25. Página de información de un proceso del proyecto

La información y estado de los documentos del proceso presenta una lista con los documentos registrados para el proceso y el presente proyecto. Para cada uno de los documentos presenta la siguiente información:

- Código del documento: Aparecerá el estado “Pendiente” si todavía no se ha introducido el código del documento.
- Nombre del documento
- Estado de la lista de comprobación del documento
- Estado de las incidencias del documento.
- Acciones para el documento: Se podrá editar o eliminar el documento. Esta opción aparecerá únicamente para los usuarios con el rol de Administradores o Responsables de Verificación.

3.2.5.1. Lista de comprobación del proceso

Al pulsar sobre el icono correspondiente a la lista de comprobación del proceso, aparecerá el contenido de dicha lista, con los datos que se hayan introducido para la misma.

En el caso de que el rol del usuario sea administrador, responsable de verificación o Verificador, se podrán editar las columnas "¿Cumple" y "Comentarios" (ver Figura 26). En este caso aparecerán los botones "Aceptar" y "Cancelar" para almacenar o no los cambios producidos.

En el caso de que el rol del usuario sea desarrollador o lector, no se podrá editar ninguna columna (ver Figura 27).

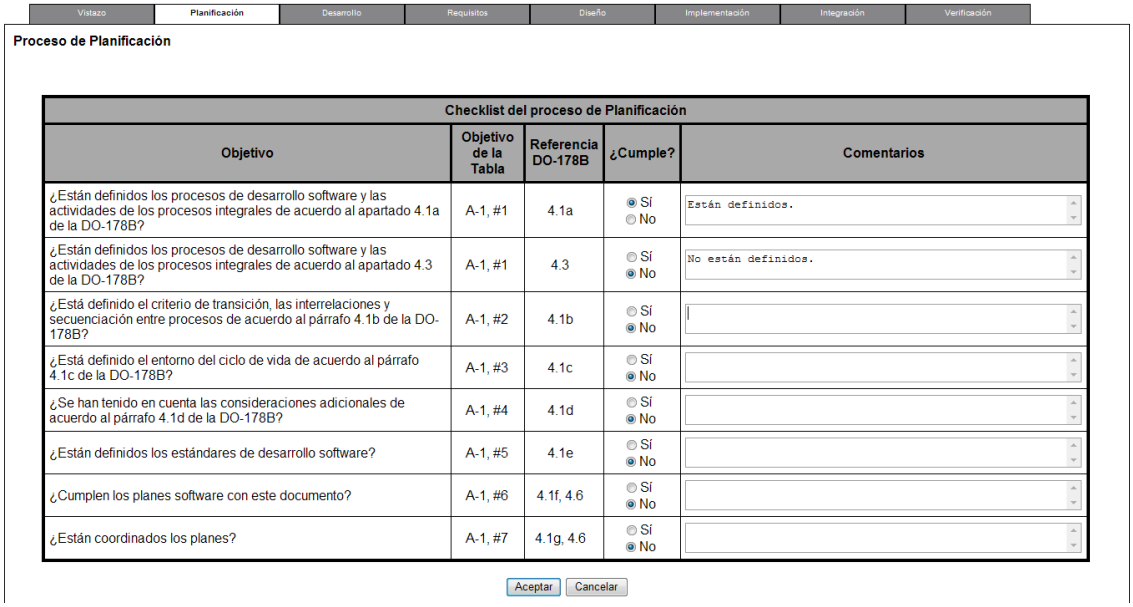


Figura 26. Lista de comprobación de un proceso – escritura

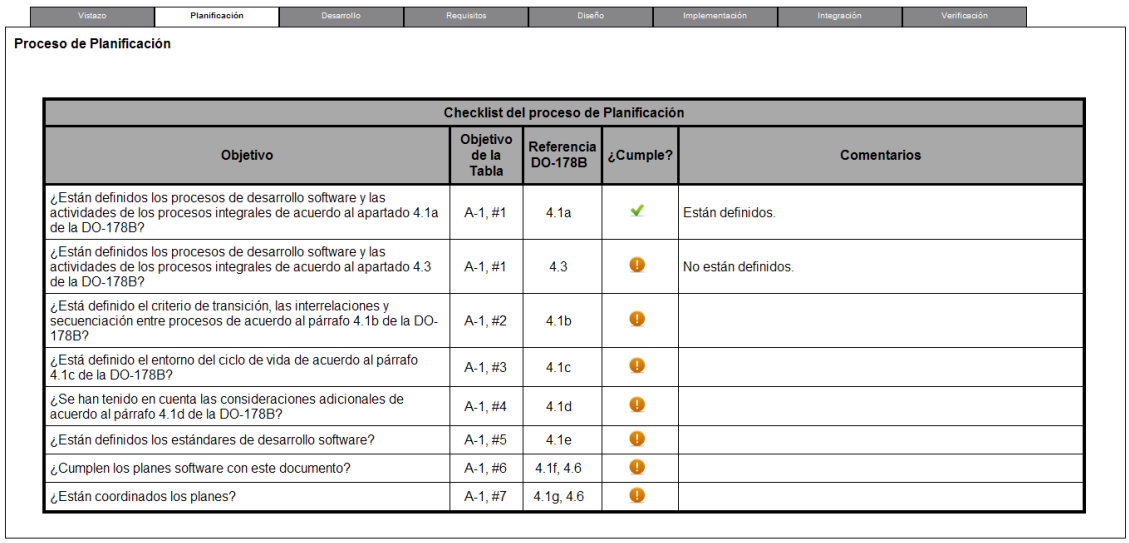


Figura 27. Lista de comprobación de un proceso – lectura

La tabla que aparece está divida en cinco columnas:

- Objetivo: Contiene la pregunta relacionada con el objetivo de la normativa aplicable.
- Objetivo de la tabla: Contiene una referencia a la tabla de la normativa y al número de objetivo de la misma.
- Referencia a la normativa: Es una referencia al apartado de la normativa que describe con más detalle el objetivo a cumplir.
- ¿Cumple?: Es la respuesta (SI/NO) para la pregunta del objetivo.
- Comentarios: Permite introducir comentarios a la respuesta, como por ejemplo el porqué no se cumple el objetivo o la localización en los documentos de dónde está contemplado el objetivo.

3.2.5.2. Lista de comprobación de un documento

Al pulsar sobre el icono correspondiente a la lista de comprobación de un documento, aparecerá el contenido de dicha lista, con los datos que se hayan introducido para la misma.

Al igual que con las listas de comprobación de los procesos, sólo los usuarios con rol "Administrador", "Responsable de Verificación" y "Verificador" podrán editar la lista.

El aspecto de una lista de comprobación para un documento es ligeramente distinta a la de un proceso.

Checklist del documento Plan de los Aspectos de Certificación Software			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Visión general del Sistema	11.1a		
¿Existe y está completa la información acerca de la visión general del sistema? Esta información debe incluir la asignación de funcionalidades que aplican al hardware y al software, la arquitectura, los procesadores utilizados, los interfaces hardware/software, y las consideraciones de safety. Si el sistema es únicamente software, se debe indicar "No aplicable. Este proyecto es para un componente de sistema"		✓	existe y está completa
Visión general del software	11.1b		
¿Existe y está completa la información acerca de la visión general del software? Esta información debe incluir las consideraciones de safety para el software, y otras consideraciones como compartición de recursos, redundancia, software disimil multiversión, toleracia ante fallos y estrategias de planificación y temporización.		✓	Existe
Consideraciones para la certificación	11.1c		
¿Está identificado y justificado el nivel de safety del software?		!	
¿Hay una descripción de las actividades de certificación para cada uno de los siguientes elementos? - Documentación de requisitos - Plan de Verificación - Plan de configuración y Plan de Calidad - Procedimientos de pruebas - Resultados de las pruebas - Matrices de trazabilidad - Índice de Configuración Software (SCI) - Índice de Cumplimiento Software (SAS) - Métodos alternativos de cumplimiento		!	
Ciclo de Vida Software	11.1d		
¿Está definido el ciclo de vida software?		!	
Datos del ciclo de vida	11.1e		

Figura 28. Lista de comprobación de un documento

En este caso, las listas de comprobación tienen la siguiente información:

- Objetivo

- Referencia a la normativa
- ¿Cumple?
- Comentarios de Verificación

Estas columnas tienen el mismo significado que en el caso de las listas de comprobación. No aparece la columna “objetivo de la tabla” debido a que la normativa no especifica los objetivos por documento, sino por proceso. No obstante, las listas de comprobación para los documentos son de utilidad para conocer el estado de cada uno de los elementos de configuración que forman parte del proceso.

En el caso de que el usuario tenga los permisos adecuados, puede modificar los comentarios, confirmándolos con el botón “Aceptar” o descartarlos con el botón “Cancelar”.

3.2.5.3. Incidencias asociadas a un documento

Cuando se pulsa en el icono de estado de las incidencias de un documento, aparece la lista de comentarios asociados al mismo.


Insertar un nuevo comentario

Plan de los Aspectos de Certificación Software							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
1	US1	A/0	comentario 1	se ha corregido	A/1	Cerrada	
2	US1	A/0	comentario 2	corregido	A/1	Cerrada	
3	US1	A/0	comentario 2	corregido	A/1	Cerrada	
4	US1	A/0	comentario 2	corregido	A/1	Cerrada	
5	US1	A/0	comentario 3	se ha corregido.	A/2	Cerrada	
6	US1	A/2	otro comentario actualizado	contestación al comentario	A/3	Cerrada	
8	US1	A/1	uno más actualizado	nada más que añadir		Cerrada	
11	US1	A/1	otra prueba	nada		Abierta	

Mostrando registros 1-8 de 8
Resultados por página: 10
Exportar a csv

Figura 29. Incidencias asociadas a un documento

La información se presenta dividida en las siguientes columnas:

- #: Número de comentario. Es único dentro de la herramienta.
- Rev.: Iniciales del revisor
- Edición: Edición del elemento de configuración objeto del comentario
- Comentario de Verificación: Texto que hace describe el comentario.
- Respuesta: Respuesta por parte del equipo de desarrollo
- Cerrado en Versión: Edición o versión del elemento de configuración en la que se da por cerrado el comentario
- Estado: Estado del comentario (Abierto/Cerrado)
- Acciones: Editar comentario o Eliminar comentario

Los datos incluidos en la columna “Revisor”, tienen forma de hiperenlace, de tal forma que es posible conocer los datos del usuario tales como nombre completo y correo electrónico pinchando en el enlace.

En el pie de página aparece información acerca de la cantidad de registros mostrados por pantalla.

También aparece la opción de presentar un determinado número de registros por página. Esta opción sólo aparece cuando el número de registros para el documento supera el número de registros por página. Por ejemplo, para menos de 10 comentarios, no aparece esta opción, pero si hay más de 10 se puede elegir que se muestren 50 comentarios por página.

Asimismo, se pueden exportar la lista de comentarios a un archivo .csv (Comma Separated Values), para poder importarlo en un archivo Excel y trabajar con él.

3.2.5.3.1. Insertar un nuevo comentario

Los usuarios que pueden insertar un comentario son los que tengan el rol “Administrador”, Responsable de Verificación” y “Verificador”.

Para insertar un nuevo comentario, basta con pulsar en la opción disponible al inicio de la tabla. Una vez seleccionada esta opción, aparecerá una nueva línea al final de la tabla para insertar los datos del nuevo comentario.

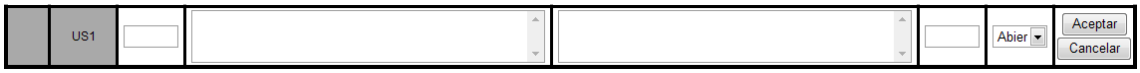


Figura 30. Insertar un nuevo comentario

Antes de insertar el nuevo comentario, éste no tiene número asignado, ya que el número lo asigna la propia herramienta en el momento de almacenarlo en la base de datos. El valor del campo “Revisor” se rellena automáticamente con las iniciales del usuario que está insertando el comentario.

Los valores mínimos que se deben introducir a la hora de crear el comentario son: “Edición” y “Comentario de Verificación”. El resto de campos son opcionales.

Para almacenar el comentario en la base de datos, basta con pulsar en el botón “Aceptar”. Si se quieren descartar los datos introducidos, se deberá pulsar el botón “Cancelar”.

3.2.5.3.2. Editar un comentario

Para editar un documento es necesario pulsar en el icono de edición que aparece en la columna “Acciones” de cada comentario.

Los usuarios que pueden editar un comentario son los que tengan el rol “Administrador”, Responsable de Verificación”, “Verificador” y “Desarrollador”. Dependiendo de este rol, podrán editar algunos campos u otros.

El administrador y el responsable de verificación podrán modificar todos los campos excepto el número de comentario y el Revisor.

Figura 31. Editar un comentario – Administrador y Responsable de Verificación

El Verificador podrá editar los mismos campos del comentario excepto la versión del documento en la que se detectó el comentario.

Figura 32. Editar un comentario – Verificador

El desarrollador únicamente podrá editar el campo “Respuesta”.

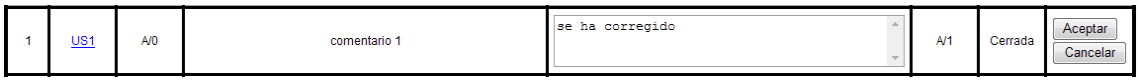


Figura 33. Editar un comentario – Desarrollador

Por último, el lector, no tendrá permisos de edición de un documento.

3.2.5.3.3. Eliminar un comentario

Para eliminar un comentario, es necesario pulsar el icono de eliminar (un aspa roja) del comentario correspondiente. Sólo Administradores, Responsables de Verificación y Verificadores tienen los permisos necesarios para eliminar un comentario.

Cuando se pulsa el botón para eliminar el comentario, aparecerá otro botón (“Borrar”) para la confirmación del borrado.

3.3. ENTORNO DE DESARROLLO Y EJECUCIÓN

En este apartado se describirá el entorno que se ha utilizado para el desarrollo de la aplicación, así como el entorno necesario para la ejecución de la misma.

3.3.1. Entorno de desarrollo

Para las actividades de desarrollo se han utilizado las siguientes herramientas:

Tabla 36. Entorno de Desarrollo

Herramienta	Versión	Descripción
NetBeans IDE	7.0.1 (Build 201109201739)	Entorno de desarrollo php y html
xampp	1.7.7	Servidor web local
PHP	5.3.8	Lenguaje interpretado

Se ha utilizado NetBeans por ser de libre distribución, y por proporcionar un entorno de desarrollo centralizado para php y html.

XAMPP es un servidor independiente de plataforma, que consiste en una base de datos MySQL, un servidor web Apache y los intérpretes para PHP y Perl.

3.3.2. Entorno de ejecución

Al estar escrita en php y html, no es necesario tener instalado ningún programa en el ordenador aparte del propio navegador de internet. Esto facilita la accesibilidad para todos los usuarios así como la disponibilidad en red de toda la información que se almacene en la base de datos.

CAPÍTULO 4: CASO PRÁCTICO

Para comprobar la viabilidad de la herramienta, en este capítulo se presentará un caso práctico real en el que se puede utilizar la herramienta, comprobando la utilidad de la herramienta así como los procesos de verificación cubiertos para la normativa aplicable.

El presente capítulo contiene una descripción del proyecto, y a continuación una descripción del sistema y del software que contiene el sistema.

El alcance de la herramienta cubre la verificación del software, por lo que el caso práctico se centrará en la verificación software del sistema.

Al tratarse de un proyecto militar y real, se han eliminado todas las referencias a datos concretos del alcance del proyecto, tales como el nombre de la aeronave.

4.1. DESCRIPCIÓN DEL PROYECTO

El proyecto tiene como alcance la sustitución de las pantallas de visualización para los pilotos en las aeronaves de transporte militar de un ejército europeo.

Estas pantallas de visualización se denominan Display Units (DUs) según los manuales de vuelo. Como este es el nombre que se utiliza para el sistema dentro de la aeronave, se utilizará este nombre a partir de ahora en el presente documento.

4.1.1. Descripción del sistema

Los Display Units (DUs) proporcionan los datos de vuelo y navegación a presentar a la tripulación (piloto/copiloto/navegante) mediante dos tipos de presentaciones: la Presentación Primaria de Vuelo (PFD) y la Presentación Secundaria (SFD). Estos datos incluyen parámetros de vuelo, datos de navegación, avisos/advertencias del sistema, datos de radiocomunicaciones, avisos del sistema MilACAS, información del radar meteorológico y otros parámetros del avión.

Los Display Units forman parte del Subsistema de Control y Presentación (CDS) del Sistema Integrado de Control y Gestión de Vuelo (SIGCV) instalado en estas aeronaves de transporte militar. El subsistema CDS contiene, entre otros, los siguientes elementos hardware:

- 2 Computadores de Misión (MCs)
- 5 Display Units (DUs) de 6x8” con pantalla AMLCD
- 3 Unidades de Control y Presentación (CDUs)
- 1 Unidad de Transferencia de Datos y Restauración de Software (MLV/DTU).

También forman parte del subsistema CDS, diferentes paneles de control situados en el tablero de instrumentos de la cabina de las aeronaves, a través de los cuales, la tripulación puede seleccionar y controlar la presentación en las pantallas. El subsistema CDS se comunica con el resto de los subsistemas de aviónica de la aeronave a través de un bus MIL-STD-1553B redundante y otros tipos de buses de datos serie (ARINC 429, ARINC 708/453, etc).

Las unidades de presentación de datos de vuelo (DUs) forman parte del Sistema de Instrumentos Electrónicos de Vuelo (EFIS). El EFIS proporciona el máximo grado de integración en la presentación de los datos de los sensores, subsistemas de navegación, control de vuelo, radar meteorológico y subsistemas de comunicaciones, proporcionando a esta información el máximo grado de flexibilidad y redundancia.

Existen cinco (5) DUs idénticas e intercambiables, dos para el piloto y otras dos para el copiloto situadas sobre el panel principal de instrumentos, y una quinta en la estación del navegante. Las 5 DUs están configuradas como dos conjuntos independientes, estando las pantallas DU1-DU2 (piloto) y DU5 (navegante) bajo control primario del MC No. 1 y DU3-DU4 (copiloto) bajo control primario del MC No. 2. Cuando sólo se dispone de un MC, éste controlará todas las DUs.

Presentan sobre una pantalla multicolor de 6x8" de cristal líquido de matriz activa (AMLCD) de alta resolución (480×640 píxeles), el texto y la simbología asociados con la Presentación Primaria de Vuelo (PFD) y la Presentación Secundaria de Vuelo (SFD). Estos displays presentan los parámetros de vuelo, datos de navegación, avisos/advertencias del sistema, datos de radiocomunicaciones, avisos del sistema MilACAS, información del radar meteorológico en colores saturados (verde, rojo y amarillo) y otros parámetros del avión.

Las DUs son compatibles con Sistemas de Visión Nocturna (NVIS tipo I y II, clase B).

Las DUs utilizan ventilación forzada proporcionada por unos ventiladores externos (uno por unidad) situados debajo de cada DU. El funcionamiento de cada ventilador es monitorizada por

un detector de baja velocidad (uno por ventilador) que envían una señal de aviso a los MCs cuando falla alguno de los ventiladores.

Las DUs disponen de un sistema de iluminación interno (backlight) que permite aumentar o disminuir el brillo del display. El control del brillo se realiza, bien por presión directa sobre el botón de ajuste que incorporan las pantallas, bien desde los controles de los equipos Display Control Panel (DCP) del piloto, copiloto y navegante.

4.1.2. Descripción del Software

La DU dispone de un único elemento de software, el OFP (Operational Flight Program), cuya carga y restauración se controla a través del MC. El OFP de la DU realiza las siguientes funciones:

1. Generación y presentación de todos los símbolos y texto necesarios para generar las presentaciones PFD y SFD, incluidos los símbolos específicos de la presentación radar.
2. Vigilancia del estado de operatividad de las DUs, BIT (POBIT, MBIT y CBIT), control de redundancia y aviso de fallos.
3. Funciones de control del hardware de la DU: control y selección de los enlaces de datos particularizados ARINC 429, control del cristal líquido (drivers de líneas y columnas, etc.), control de la intensidad y consistencia de la iluminación posterior, interfase del circuito de indicación de tiempo de funcionamiento (ETI: Elapsed Time Indicator), ajuste del brillo ordenado por el MC, control del ventilador, etc.
4. Gestión de la restauración y verificación del OFP de las DUs; el OFP se introduce en el SICGV vía DTU/MLV (en su función MLV) y se retransmite a las DUs a través de los MCs.
5. Gestión de datos de entrada y salida.

4.2. APLICACIÓN DE LA HERRAMIENTA EN EL CASO PRÁCTICO

4.2.1. Nivel del Software

Para demostrar que el software fue diseñado, verificado y validado correctamente, se aplicaron las consideraciones proporcionadas en el documento RTCA/DO-178B como un medio aceptable para asegurar la aprobación del software. En el Estudio Funcional de los Riesgos del Sistema (FHA) se concluyó que el nivel de aseguramiento para el software debía ser DAL-B.

4.2.2. Ciclo de vida software

Este proyecto siguió un ciclo de vida en “V”, y se identificaron los mismos procesos definidos en la DO-178B:

- Planificación Software
- Análisis de requisitos software
- Diseño software
- Codificación y pruebas de CSU
- Integración y pruebas de CSC
- Integración y pruebas del sistema software

Las salidas que se obtuvieron de esta fase se resumen en la siguiente figura:

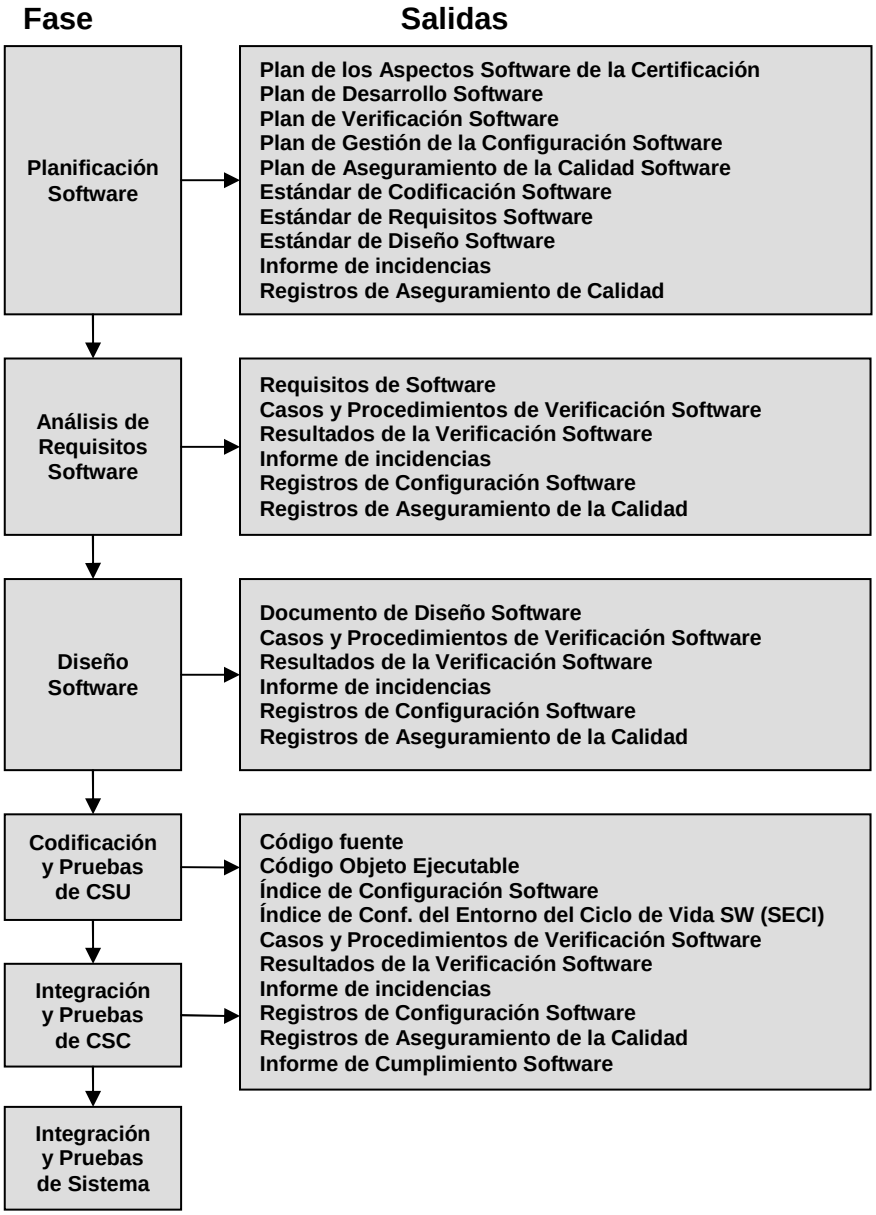


Figura 34. Fases y Salidas del Ciclo de Vida Software

Para cada una de estas fases y las salidas del ciclo de vida, se puede utilizar la herramienta.

En los siguientes apartados se muestra cómo se aplica AVeMaCS al ciclo de vida.

Para las siguientes tablas presentadas, es necesario hacer notar que si la Edición/Revisión del documento aparece marcada con un asterisco, es para indicar que se revisó antes de cerrar dicha Edición/Revisión, y se comprobó si se había corregido o no la incidencia correspondiente una vez cerrada.

4.2.2.1. Proceso de planificación

Para el proceso de planificación se generaron los siguientes documentos:

Tabla 37. Documentos generados en el proceso de Planificación

Código	Nombre	Referencia DO-178B
0149900000000PL01	Plan de Certificación Software (PSAC)	11.1
0149900000000PD00	Plan de Desarrollo Software (SDP)	11.2
0149900000000PL02	Plan de Verificación Software (SVP)	11.3
0149900000000PL03	Plan de Configuración Software (SCMP)	11.4
0149900000000PL04	Plan de Calidad Software (SQAP)	11.5
0149900000000ST00	Estándar de Especificación Software (SRS)	11.6
0149900000000ST01	Estándar de Diseño Software (SDS)	11.7
0149900000000ST02	Estándar de Codificación Software (SCS)	11.8

El resultado final de la lista de comprobación para el proceso de planificación se muestra en la Figura 35.

Checklist del proceso de Planificación				
Objetivo	Objetivo de la Tabla	Referencia DO-178B	¿Cumple?	Comentarios
¿Están definidos los procesos de desarrollo software y las actividades de los procesos integrales de acuerdo al apartado 4.1a de la DO-178B?	A-1, #1	4.1a	✔	PSAC, SDP, SVP, SCMP, SQAP
¿Están definidos los procesos de desarrollo software y las actividades de los procesos integrales de acuerdo al apartado 4.3 de la DO-178B?	A-1, #1	4.3	✔	PSAC, SDP, SVP, SCMP, SQAP
¿Está definido el criterio de transición, las interrelaciones y secuenciación entre procesos de acuerdo al párrafo 4.1b de la DO-178B?	A-1, #2	4.1b	✔	PSAC, SDP, SVP, SCMP, SQAP
¿Está definido el entorno del ciclo de vida de acuerdo al párrafo 4.1c de la DO-178B?	A-1, #3	4.1c	✔	SDP
¿Se han tenido en cuenta las consideraciones adicionales de acuerdo al párrafo 4.1d de la DO-178B?	A-1, #4	4.1d	✔	PSAC, SDP, SVP, SCMP, SQAP
¿Están definidos los estándares de desarrollo software?	A-1, #5	4.1e	✔	SRS, SDS, SCS
¿Cumplen los planes software con este documento?	A-1, #6	4.1f, 4.6	✔	Registros del Aseguramiento de la Calidad, Resultados de la Verificación Software
¿Están coordinados los planes?	A-1, #7	4.1g, 4.6	✔	Registros del Aseguramiento de la Calidad, Resultados de la Verificación Software

Figura 35. Resultado de la lista de comprobación del proceso de planificación

El resultado de las listas de comprobación para cada uno de los documentos pertenecientes al proceso de planificación se presenta en las siguientes sub-apartados.

4.2.2.1.1. PSAC

El resultado de la lista de comprobación para el PSAC se muestra en la Figura 36.

Checklist del documento Plan de Certificación Software (PSAC)			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Visión general del Sistema	11.1a		
¿Existe y está completa la información acerca de la visión general del sistema? Esta información debe incluir la asignación de funcionalidades que aplican al hardware y al software, la arquitectura, los procesadores utilizados, los interfaces hardware/software, y las consideraciones de safety. Si el sistema es únicamente software, se debe indicar "No aplicable. Este proyecto es para un componente de sistema"		✔	Toda esta información está contenida en el apartado 3 del documento.
Visión general del software	11.1b		
¿Existe y está completa la información acerca de la visión general del software? Esta información debe incluir las consideraciones de safety para el software, y otras consideraciones como compartición de recursos, redundancia, software disímil multiversión, tolerancia ante fallos y estrategias de planificación y temporización.		✔	La información está contenida en el apartado 4 del documento.
Consideraciones para la certificación	11.1c		
¿Está identificado y justificado el nivel de safety del software?		✔	De acuerdo al Estudio Funcional de Riesgos (FHA), el nivel de aseguramiento del software debe ser DAL-B, y así se referencia en
¿Hay una descripción de las actividades de certificación para cada uno de los siguientes elementos? - Documentación de requisitos - Plan de Verificación - Plan de configuración y Plan de Calidad - Procedimientos de pruebas - Resultados de las pruebas - Matrices de trazabilidad - Índice de Configuración Software (SCI) - Índice de Cumplimiento Software (SAS) - Métodos alternativos de cumplimiento		✔	Las salidas del ciclo de vida software están detalladas en el apartado 8 del documento.
Ciclo de Vida Software	11.1d		
¿Está definido el ciclo de vida software?		✔	El ciclo de vida software y sus procesos están definidos en el apartado 7 del documento.
¿Está definido cómo se van a satisfacer cada uno de los objetivos del ciclo de vida software?		✔	Está definido dentro de los procesos del ciclo de vida software (apartado 7) y con las salidas especificadas en el apartado 8 d
¿Está definida la organización y las responsabilidades dentro del ciclo de vida?		✔	La organización y responsabilidades están detalladas en el apartado 6 del documento.
Datos del ciclo de vida	11.1e		
¿Están definidos los datos del ciclo de vida que van a ser producidos?		✔	Las salidas del ciclo de vida software están detalladas en el apartado 8 del documento.
¿Están los datos del ciclo de vida correctamente relacionados con las actividades definidas?		✔	Los datos del ciclo de vida están trazadas a las distintas fases del mismo, de las que se identifican las actividades.
Planificación	11.1f		
¿Está definida la planificación de las actividades del ciclo de vida de desarrollo?		✔	Se hace una referencia a la planificación de las actividades de desarrollo software en el apartado 9 del documento.
Consideraciones adicionales	11.1g		
¿Se han tenido en cuenta las siguientes consideraciones adicionales? - Métodos alternativos de cumplimiento - Calificación de herramientas - Software previamente desarrollado - Software COTS - Software Disímil Multiversión - Información de historia en servicio		✔	Las consideraciones adicionales se encuentran en el apartado 10 del documento.

Figura 36. Resultado de la lista de comprobación para el PSAC

Las incidencias que se registraron en el PSAC son las que se muestran en la Figura 37.

Plan de Certificación Software (PSAC)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
30	JEB	A/0*	Revisar la documentación aplicable y de referencia	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
31	JEB	A/0*	Debería poner "Interfaz para el 232" en vez de "Interfaz para 323".	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
32	JEB	A/0*	En el apartado 6.3 (Contribuciones Potenciales del Software a las condiciones de fallo), se debería clarificar que los fallos que se presentan a continuación son los que aparecen en el FHA.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
33	JEB	A/0*	En la Figura 7.1-1, el término "Registro del Plan de Calidad" debería ser sustituido por "Registros de Aseguramiento de la Calidad", y el término "Registro del Plan de Configuración Software" debería ser sustituido por "Registros de la Configuración Software"	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
34	JEB	A/0*	En la Tabla 8-1, definir más claramente el significado de "Disponibilidad".	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
35	JEB	A/0*	En la Tabla 8-1, clarificar las etapas o Fases del Ciclo de Vida Software.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
36	JEB	A/0*	Utilizar "Software disímil multiversión" en vez de "Multiple-Version Dissimilar Software"	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
37	JEB	A/0*	Debería incluirse un apartado que hable de las herramientas de desarrollo y validación, y de la calificación de las mismas (si procede), según el apartado 11.1g de la DO-178B.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
38	JEB	A/1	No se han encontrado incidencias en esta Edición/Revisión del documento.	No son necesarios cambios en esta Edición/Revisión del documento.	N/A	Cerrada	

Figura 37. Comentarios registrados para el PSAC

Los comentarios registrados para el PSAC se refieren principalmente a la inclusión de aclaraciones adicionales a algunos contenidos. Lo más importante hace referencia a la necesidad de incluir referencias a las herramientas de desarrollo y verificación, y a la calificación de las mismas en caso de que procediese.

4.2.2.1.2. SDP

Checklist del documento Plan de Desarrollo Software (SDP)			
Objetivo	Referencia DO-178B	¿Cumple?	Comentarios Verificación
Estándares	11.2a		
¿Se han identificado los estándares de requisitos, diseño y codificación?		✔	El Estándar de Especificación Software (0149900000000ST00), Estándar de Diseño Software (0149900000000ST01) y Estándar de C
¿Se han identificado los estándares para el software previamente desarrollado, incluyendo los COTS?		✔	No existe software previamente desarrollado.
Ciclo de Vida Software	11.2b		
¿Se han definido los procesos del ciclo de vida software?		✔	Apartado 4 (Ciclo de vida Software)
¿Se ha definido el criterio de transición?		✔	El criterio de transición entre fases del ciclo de vida se define dentro del ciclo de vida software (Apartado 4)
Entorno de desarrollo software	11.2c		
¿Se ha definido el entorno de desarrollo software?		✔	El entorno de desarrollo se define en el apartado 5 (Entorno de Desarrollo Software)
¿Se ha definido el método y herramientas para el desarrollo de requisitos?		✔	Las herramientas de desarrollo de requisitos están definidas en el apartado 5.1. El método de desarrollo está en el Estándar
¿Se ha definido el método y herramientas para el diseño?		✔	Las herramientas de diseño están definidas en el apartado 5.2. El método de desarrollo está en el Estándar de Diseño Softw
¿Se ha definido el lenguaje de programación, las herramientas de desarrollo y el compilador a utilizar?		✔	Toda esta información está definida en el apartado 5.3.
¿Se ha identificado la plataforma hardware que se va a utilizar?		✔	La plataforma HW está definida en el apartado 5.5.

Figura 38. Resultado de la lista de comprobación para el SDP

Plan de Desarrollo Software (SDP)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
39	JEB	A/0*	Se debería sustituir la frase, "Este plan de desarrollo está relacionado con los siguientes planes" por "Este plan de desarrollo está relacionado con los siguientes planes y estándares".	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
40	JEB	A/0*	Sería necesario revisar las referencias en los documentos aplicables, y poner sólo las que aporten información al SDP.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
41	JEB	A/0*	En la Figura 4-1, el término "Registro del Plan de Calidad" debería ser sustituido por "Registros de Aseguramiento de la Calidad", y el término "Registro del Plan de Configuración Software" debería ser sustituido por "Registros de la Configuración Software"	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
42	JEB	A/0*	De acuerdo con el SVP, los planes de software se verificarán mediante revisiones y/o análisis, no mediante pruebas. Se debería especificar lo mismo que en el SVP.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
43	JEB	A/0*	La Tabla 4.2-1 debería hacer referencia al Plan de Gestión de Proyecto, en vez de al Priego de Prescripciones Técnicas.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
44	JEB	A/0*	El estándar que se va a utilizar es sólo para C, no para ensamblador.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
45	JEB	A/0*	La plataforma HW para el proceso de análisis de requisitos del SW debería ser un PC (donde se instala la aplicación Requisite Pro) y un servidor corporativo, donde se almacenan los documentos y la BBDD.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
46	JEB	A/0*	La plataforma HW para el proceso de análisis de requisitos del SW debería ser un PC (donde se instala la aplicación Requisite Pro) y un servidor corporativo, donde se almacenan los documentos y la BBDD.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
47	JEB	A/0*	Falta el título de la Tabla 5-1.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
48	JEB	A/0*	En la tabla 5-1, debería especificarse que se va a utilizar la herramienta Cantata++ para realizar las pruebas unitarias.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
49	JEB	A/0*	Según el apartado 4.1a de la DO-178B, habría que incluir una referencia al nivel B, con el que se va a desarrollar el software.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
50	JEB	A/0*	En el apartado 5, debería aparecer una breve descripción de la herramienta de control de configuración o hacer una referencia al SCMP si contiene esta información.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
51	JEB	A/1	No se han encontrado incidencias en esta Edición/Revisión del documento.	No son necesarios cambios en esta Edición/Revisión del documento.	N/A	Cerrada	

Figura 39. Comentarios registrados para el SDP

En el caso del SDP, se indicó que era necesario incluir alguna información requerida por la DO-178B (Ref. [4]), como por ejemplo el nivel aplicable para el software. Algunos otros comentarios fueron relacionados con el formato o información adicional para mejorar el contenido del documento.

4.2.2.1.3. SVP

Checklist del documento Plan de Verificación Software (SVP)			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Organización	11.3a		
¿Está definida la organización dentro del proceso de verificación y los interfaces con el resto de los procesos del ciclo de vida?		✔	La organización está definida dentro del apartado 3.1, y las responsabilidades dentro del apartado 3.3.
Independencia	11.3b		
¿Está definida la independencia del proceso de verificación (si es necesaria)?		✔	La independencia está definida dentro del apartado 3.2 del plan
¿Es suficiente el nivel de dependencia especificada para los procesos de verificación?		✔	La independencia definida cubre con los criterios de la DO-178B para un nivel B de aseguramiento.
Métodos de Verificación	11.3c		
¿Están definidos los métodos de verificación para cada una de las actividades del proceso de verificación?		✔	Los métodos de verificación están definidos en el apartado 4.3.2 del plan.
Entorno de Verificación	11.3d		
¿Están definidos los equipos de pruebas, herramientas de análisis y herramientas para las pruebas que se van a utilizar? En caso de que alguno de estos elementos esté descrito en otro documento, deberá aparecer una referencia al mismo.		✔	El entorno de verificación se encuentra definido en el apartado 4.3.1 del plan.
¿Está definido el entorno hardware de pruebas? En caso de que el entorno se defina en otro documento, debe aparecer una referencia al mismo.		✔	La referencia al entorno hardware de pruebas se encuentra dentro del apartado 4.3.1.5.
Criterio de Transición	11.3e		
¿Está definido el criterio de transición para llevar a cabo las tareas de verificación?		✔	El criterio de transición está definido en el apartado 4.2.
Consideraciones acerca de las particiones	11.3f		
En el caso de que existan particiones, ¿están definidos los métodos para verificar la integridad de las mismas?		✔	No existen particiones software.
Suposiciones acerca del compilador	11.3g		
¿Existe una descripción de las suposiciones acerca del compilador?		✔	Las suposiciones acerca del compilador están descritas en el apartado 7.2.
Estrategia de re-verificación	11.3h		
¿Está definida la estrategia de re-verificación cuando existan modificaciones en el software?		✔	Las directrices de re-verificación están descritas en el apartado 5.10
Software previamente desarrollado	11.3i		
En el caso de que exista software previamente desarrollado que no cumpla los objetivos de la normativa aplicable, ¿está descrita la forma en la que se van a satisfacer los objetivos de la normativa aplicable?		✔	No existe software desarrollado con anterioridad, tal y como se establece en el apartado 7.3
Software disímil multiversión	11.3j		
Si se utiliza la técnica de software disímil multiversión, ¿Están descritos los procesos de verificación en este caso?		✔	No se utiliza la técnica de software disímil multiversión, tal y como indica el apartado 7.4

Figura 40. Resultado de la lista de comprobación para el SVP

Plan de Verificación Software (SVP)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
81	JEB	A/0*	En la página 1-2, la referencia del PSAC no queda clara. Habría que clarificar la frase.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
82	JEB	A/0*	En la página 1-3, en el apartado de "Relación con otros planes", debería incluirse el Plan de Integración y Pruebas.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
83	JEB	A/0*	En la página 4-3, en la Tabla 4.2-1 hay que incluir en "Entradas" La Especificación de los Requisitos del Sistema (SSS), ya que es el documento que se utiliza para trazar los requisitos software desde los requisitos de sistema.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
84	JEB	A/0*	En la página 4-4, en la Tabla 4.3-1, habría que incluir en las Entradas el Estándar de Diseño Software, puesto que este documento también es entrada en la verificación del proceso de diseño software.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
85	JEB	A/0*	En la página 4-5, en la tabla 4.4-1, hay que incluir como entradas el Documento de Diseño Software, ya que también es entrada del proceso de verificación de la codificación e integración.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
86	JEB	A/0*	En la página 4-7, como la planificación del proceso de verificación software está incluida dentro de la planificación general del proyecto, es necesario hacer referencia al "Plan de Gestión de Proyecto", de tal forma que se evitan inconsistencias entre los planes y se mejora la mantenibilidad del documento.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
87	JEB	A/0*	En la página 5-2, en el apartado "Independencia" se debería eliminar la figura con la organización del programa, ya que está previamente incluida en el apartado de "Organización".	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
88	JEB	A/0*	En la página 5-1, debería hacerse más énfasis en la explicación de la organización independiente para la validación y verificación según la DO-178B.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
89	JEB	A/0*	En la página 6-1, debería estructurarse mejor el apartado de "Métodos de Verificación", incluyendo las actividades que se van a realizar utilizando cada método.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
90	JEB	A/0*	Se debe especificar la estrategia y métodos para la realización de las pruebas unitarias.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	

Figura 41. Comentarios registrados para el SVP (I)

Plan de Verificación Software (SVP)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
91	JEB	A/0*	En la página 7-1, se debe revisar y actualizar la Tabla con las herramientas utilizadas para el proceso de Verificación Software.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
92	JEB	A/0*	Falta especificar cómo se van a trazar los requisitos de alto nivel con los requisitos de bajo nivel.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
93	JEB	A/0*	No está especificado para qué tipo de pruebas se va a utilizar la herramienta Cantata++.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
94	JEB	A/0*	Es necesario especificar el equipo hardware que se va a utilizar en la realización de las pruebas o hacer referencia al documento donde esté especificado.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
95	JEB	A/0*	Debería incluirse un apartado específico con las "Consideraciones Adicionales" según la norma DO-178B.	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
96	JEB	A/0*	Debería incluirse una breve descripción del compilador utilizado o hacer referencia al documento donde esté esta descripción (SDP).	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
97	JEB	A/0*	La traducción para "Multiple Version Dissimilar Software" debería sustituirse por "Software Disímil Multiversión".	Revisado y corregido antes de cerrar la versión A/0 del documento	A/0	Cerrada	
98	JEB	A/0*	Debido a que el documento tiene un índice diferente al que aparece en la norma DO-178B, es necesario incluir una trazabilidad entre la norma y el presente documento para clarificar los contenidos.	Se ha incluido el Apéndice A con una trazabilidad de los contenidos que debe incluir el Plan de Verificación según la norma DO-178B (apartado 11.3) y los contenidos del SVP del presente proyecto.	A/0	Cerrada	
99	JEB	A/0	Es necesario corregir el número del documento "Casos y Procedimientos de la Verificación Software". Actualmente, el número es el 0149900000000AT03.	Número del documento corregido.	A/1	Cerrada	
100	JEB	A/0	Es necesario corregir el número del documento "Estándar de Codificación Software". Actualmente, el número es el 0149900000000ST02.	Número del documento corregido.	A/1	Cerrada	

Figura 42. Comentarios registrados para el SVP (II)

Plan de Verificación Software (SVP)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
101	JEB	A/0	Es necesario corregir el número del documento "Estándar de Especificación Software". Actualmente, el número es el 0149900000000ST02.	Número del documento corregido.	A/1	Cerrada	
102	JEB	A/0	Es necesario corregir el número del documento "Estándar de Diseño Software". Actualmente, el número es el 0149900000000ST01.	Número del documento corregido.	A/1	Cerrada	
103	JEB	A/0	Es necesario corregir los números de los documentos especificados en los cuatro comentarios anteriores a lo largo de todo el documento.	Números de los documentos corregidos.	A/1	Cerrada	
104	JEB	A/1	Según la conclusión acordada con la Autoridad Certificadora, deberá indicarse que el software será calificado de acuerdo a un nivel DAL B, excepto para las funciones de procesado y monitorización de datos críticos (actitud, velocidad y altitud), que serán calificadas de acuerdo a un nivel DAL A.	Incluido un párrafo aclaratorio en el apartado 1.1. Además, se ha incluído una aclaración adicional en el apartado 4.3.2.2.1.	A/2	Cerrada	
105	JEB	A/1	En el apartado 1.4 deberían incluirse los estándares, ya que forman parte del proceso de planificación.	Incluidas la referencias a los estándares de especificación, diseño y codificación software.	A/2	Cerrada	
106	JEB	A/1	Es necesario revisar la organización del proyecto de acuerdo al Plan de Gestión de Proyecto.	Revisado y actualizado según el Plan de Gestión del Proyecto.	A/2	Cerrada	
107	JEB	A/1	Falta por incluir el informe de resultados del proceso de planificación software.	Incluido.	A/2	Cerrada	
108	JEB	A/2	Es necesario actualizar el documento de acuerdo a los coomentarios de la Autoridad Certificadora: "Sólo se tiene en cuenta el nivel de SW B y no se explican las actividades que se deben/van a desarrollar para la parte del SW que se califica como nivel A."	Incluida una aclaración en el apartado 1.1, actualizadas las tablas de objetivos para nivel A (donde aplica).	A/3	Cerrada	

Figura 43. Comentarios registrados para el SVP (III)

Los comentarios acerca del SVP también están relacionados con la necesidad de incluir información adicional para completar ciertos aspectos del plan. Otro comentario está relacionado con la organización especificada, que no coincidía exactamente con la mostrada en el Plan de Gestión del Proyecto. Algunos de los comentarios se incluyeron como respuesta a los comentarios de la Autoridad Certificadora.

4.2.2.1.4. SCMP

Checklist del documento Plan de Configuración Software (SCMP)			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Entorno	11.4a		
¿Está definido el entorno de configuración software definido? Esto incluye herramientas, procedimientos, métodos, estándares, organización, responsabilidades y los interfaces entre los procesos definidos.		✔	El entorno de configuración software está definido en el apartado 3, e incluye la organización, las responsabilidades, los pr
Actividades	11.4b		
¿Existe una descripción para cada una de las actividades del proceso de configuración software de los siguientes?		✔	Sí, una referencia al contenido se indica en las siguientes respuestas de la lista de comprobación.
¿Está definida la identificación de la configuración? Elementos a ser identificados, cuándo van a ser identificados, los métodos de identificación para los elementos de configuración y la relación de la identificación con el sistema.		✔	La identificación de la configuración está incluída en el apartado 4.1
¿Están descritas las líneas base y la trazabilidad? Cómo se establecen las líneas base, cuándo serán establecidas, y la trazabilidad de las líneas base con los elementos de configuración.		✔	El establecimiento de las líneas de referencia está descrito en el apartado 4.2
¿Está descrito el método para los informes de problemas? Contenido e identificación de los mismos, cuándo son escritos, criterio y método para cerrarlos, y su relación con la actividad de control de cambios.		✔	El método para los informes de incidencias está incluído en el apartado 4.3
¿Está descrita la actividad de control de cambios? Elementos de configuración y líneas base a ser controlados, cuándo serán controlados y métodos para preservar la integridad de las líneas base y los elementos de configuración.		✔	El control de configuración y control de cambios está descrito en el apartado 4.4
¿Está descrito el método para la revisión de cambios?		✔	La revisión de los cambios está incluída en el apartado 4.4.3
¿Está descrito el método para llevar a cabo el seguimiento del estado de la configuración?		✔	El seguimiento de estado de la configuración está descrita en el apartado 4.4. El registro del estado de la configuración se
¿Están descritas las actividades de archivo, recuperación y entrega de la información bajo control de configuración?		✔	Las actividades de archivo, recuperación y entrega de versiones están incluídas en el apartado 4.5
¿Están definidos los mecanismos de control de carga?		✔	El mecanismo de control de carga del software está descrito en el apartado 4.8 del plan
¿Están definidos los controles para las herramientas de desarrollo y verificación?		✔	El control del entorno del ciclo de vida software se define en el apartado 4.9
¿Están definidos los controles asociados a las categorías de control para los elementos de configuración?		✔	El control de los datos del ciclo de vida software se describe en el apartado 4.10
Criterio de Transición	11.4c		
¿Está definido el criterio de transición para los procesos de control de configuración?		✔	El criterio de transición para las actividades del proceso de gestión de la configuración se encuentra en el apartado 5
Datos de configuración	11.4d		
¿Están definidos los informes de configuración a generar? Esto incluye SCI, SECI y registros de configuración.		✔	Las salidas de la gestión de configuración software están descritas en el apartado 5.
Control de Proveedores	11.4e		
¿Se han definido los métodos para asegurar que los sub-contratistas cumplirán con el Plan de Configuración Software?		✔	No existen suministradores externos de software, tal y como se especifica en el apartado 6 del plan.

Figura 44. Resultado de la lista de comprobación para el SCMP

Plan de Configuración Software (SCMP)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
162	JEB	A/0*	Debería incluirse un apartado para incluir la relación con otros planes y estándares, para que hubiera una consistencia de contenido entre todos.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
163	JEB	A/0*	Por consistencia con el resto de los planes, la sección "Definiciones y Acrónimos" debería ser renombrada como "NOTAS" y ser la última sección del documento.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
164	JEB	A/0*	En la página 2-1, debería ponerse completa la definición de "Línea de Referencia Asignada" que aparece en PGR-IDR-080, ya que clarificaría la definición.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
165	JEB	A/0*	En la definición de "Comité de Control de la Configuración" es necesario cambiar la palabra "derogaciones" por "concesiones", para que la definición sea más precisa.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
166	JEB	A/0*	La definición del acrónimo para PSAC no es la correcta. La correcta es "Plan de los Aspectos Software de la Certificación"	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
167	JEB	A/0*	El acrónimo utilizado para el Índice de Configuración del Entorno del Ciclo de Vida Software es SECI (en lugar de SLCECI). Revisar este acrónimo en todo el documento.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
168	JEB	A/0*	Sería necesario revisar los documentos aplicables.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
169	JEB	A/0*	El término "derogación" debe ser sustituido por "concesión" en todo el documento.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
170	JEB	A/0*	En la página 5-1, debería clarificarse y simplificarse la explicación de la identificación de versiones de los documentos y hacer referencia al Plan de Documentación.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
171	JEB	A/0*	Se debe utilizar el término "Informe de Incidencias" en lugar de "Informe de Problemas". Esto debería ser revisado en todo el documento.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	

Figura 45. Comentarios registrados para el SCMP (I)

Plan de Configuración Software (SCMP)							
#	Rev.	Edición	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
172	JEB	A/0*	En la página 5-3, es necesario poner una referencia al lugar donde se encuentra el formato de PCI.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
173	JEB	A/0*	Sustituir "liberación de Versiones" por "entrega de versiones" en el título del apartado 5.5.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
174	JEB	A/0*	Incluir un subapartado con la identificación del soporte físico del software.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
175	JEB	A/0*	En la página 8-1, poner los códigos de configuración de los documentos.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
176	JEB	A/1	No se han encontrado incidencias en esta Edición/Revisión del documento.	No son necesarios cambios en esta Edición/Revisión del documento.	N/A	Cerrada	
177	JEB	B/0	No se han encontrado incidencias en esta Edición/Revisión del documento.	No son necesarios cambios en esta Edición/Revisión del documento.	N/A	Cerrada	

Figura 46. Comentarios registrados para el SCMP (II)

El Plan de Configuración Software fue modificado para incluir comentarios relacionados con la comprensión del documento, y otros aspectos que son necesarios tener en cuenta según la normativa de la compañía.

4.2.2.1.5. SQAP

Checklist del documento Plan de Calidad Software (SQAP)			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Entorno	11.5a		
¿Está descrito el entorno para las actividades de Calidad? Esto incluye alcance, responsabilidades, estándares, procedimientos, herramientas y métodos.		✔	El entorno para las actividades de Calidad está descrito en los apartados 3 y 4 del plan.
Autoridad	11.5b		
¿Está establecida la autoridad de Calidad?		✔	Las auditorías de Calidad están definidas en el apartado 4.11
¿Está establecida la independencia y responsabilidad de las actividades de Calidad?		✔	Las organización y responsabilidades de las actividades de Calidad están descritas en el apartado 3 del plan.
Actividades	11.5c		
¿Se han definido las actividades a llevar a cabo para cada fase del ciclo de vida?		✔	Las actividades están descritas en el apartado 4
¿Se han definido los métodos para llevar a cabo las actividades de Calidad dentro del ciclo de vida?		✔	Los métodos están descritos en los apartados 4.1 a 4.10 del plan
¿se han definido las actividades a llevar a cabo con respecto a los informes de problemas?		✔	Las actividades con respecto a los informes de problemas están definidos en los apartados 4.5, 4.6 y 4.7
¿Se ha definido la "Software Conformity Review"?		✔	Está definida en el apartado 6
Criterio de Transición	11.5d		
¿Se ha definido el criterio de transición para las actividades de Calidad?		✔	El criterio de transición está definido en el apartado 5 del plan
Planificación	11.5e		
¿Se ha definido la relación temporal de las actividades de Calidad con respecto a las actividades del ciclo de vida software?		✔	Esta información está contenida en el apartado 4.2
Registros de Calidad	11.5f		
¿Se han definido los registros de Calidad que se van a generar?		✔	Los registros de Calidad a generar están definidos en el apartado 6
Control de Proveedores	11.5g		
¿Se han definido los métodos para asegurar que los sub-contratistas cumplirán con el Plan de Calidad?		✔	No existen proveedores externos para el software.

Figura 47. Resultado de la lista de comprobación para el SQAP

Plan de Calidad Software (SQAP)							
#	**Rev.**	**Edición**	**Comentario de Verificación**	**Respuesta**	**Cerrado en Versión**	**Estado**	**Acciones**
109	JEB	A/0*	Es necesario revisar la introducción.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
110	JEB	A/0*	En la página 1-1, sería necesario introducir un nuevo subapartado donde se define el ámbito del documento.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
111	JEB	A/0*	Revisar la documentación aplicable y de referencia, e incluir sólo la que corresponda.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
112	JEB	A/0*	Cambiar la palabra 'problema' por 'incidencia'.	Revisado y corregido antes de cerrar la versión A/0 del documento.	A/0	Cerrada	
113	JEB	A/0	Según la conclusión acordada con la Autoridad Certificadora, deberá indicarse que el software será calificado de acuerdo a un nivel DAL B, excepto para las funciones de procesado y monitorización de datos críticos (actitud, velocidad y altitud), que serán calificadas de acuerdo a un nivel DAL A.	Incluída esta aclaración en el apartado 1.2 del documento.	A/1	Cerrada	
114	JEB	A/2	No se han encontrado incidencias en esta Edición/Revisión del documento.	No son necesarios cambios en esta Edición/Revisión del documento.	N/A	Cerrada	

Figura 48. Comentarios registrados para el SQAP (I)

El Plan de Calidad Software no tuvo muchos comentarios, pero sí que hizo falta incluir alguna aclaración por petición de la Autoridad Certificadora. El resto de comentarios se refieren a correcciones menores que fue necesario realizar.

4.2.2.2. Proceso de desarrollo

El proceso de desarrollo se completó satisfactoriamente, y el resultado de la lista de comprobación se incluye en la siguiente figura.

Checklist del proceso de Desarrollo				
Objetivo	**Objetivo de la Tabla**	**Referencia DO-178B**	**¿Cumple?**	**Comentarios**
¿Han sido desarrollados los requisitos de alto nivel?	A-2, #1	5.1.1a	✔	Los requisitos software de alto nivel están especificados en el documento 'Especificación de Requisitos Software (SRD)'
¿Han sido definidos los requisitos derivados de alto nivel?	A-2, #2	5.1.1b	✔	No ha sido necesario definir requisitos derivados de alto nivel.
¿Ha sido desarrollada la arquitectura software?	A-2, #3	5.2.1a	✔	La arquitectura software está descrita en el 'Documento de Diseño Software (SDD)', con número 0149900000000DS00
¿Han sido desarrollados los requisitos de bajo nivel?	A-2, #4	5.2.1a	✔	Los requisitos software de bajo nivel están especificados en el 'Documento de Diseño Software (SDD)', con número 014990000000
¿Han sido definidos los requisitos derivados de bajo nivel?	A-2, #5	5.2.1b	✔	No ha sido necesario definir requisitos software de bajo nivel.
¿Ha sido desarrollado el código fuente?	A-2, #6	5.3.1a	✔	El código fuente está desarrollado y bajo control de configuración.
¿Ha sido producido el código objeto ejecutable y se ha integrado en el hardware de destino?	A-2, #7	5.4.1a	✔	Todas las pruebas realizadas están realizadas con el hardware final.

Figura 49. Resultado de la lista de comprobación del proceso de desarrollo

4.2.2.3. Proceso de requisitos

Los objetivos del proceso de requisitos se revisaron, completando la lista de comprobación mostrada en la siguiente figura.

Checklist del proceso de Requisitos				
Objetivo	**Objetivo de la Tabla**	**Referencia DO-178B**	**¿Cumple?**	**Comentarios**
¿Los requisitos software de alto nivel cumplen con los requisitos de sistema?	A-3, #1	6.3.1a	✔	Los requisitos están en el documento Especificación de Requisitos Software (SRD)
¿Los requisitos software de alto nivel son precisos y consistentes?	A-3, #2	6.3.1b	✔	Se han revisado y son precisos y consistentes.
¿Los requisitos software de alto nivel son compatible con la plataforma hardware?	A-3, #3	6.3.1c	✔	Son compatibles.
¿Los requisitos software de alto nivel son verificables?	A-3, #4	6.3.1d	✔	Sí. Además, cada requisito software de alto nivel tiene un método de verificación.
¿Los requisitos software de alto nivel cumplen con los estándares?	A-3, #5	6.3.1e	✔	Sí.
¿Los requisitos software de alto nivel están trazados con los requisitos de sistema?	A-3, #6	6.3.1f	✔	Trazabilidad SRD – SSS. Verificado mediante las reglas RULE_TR_1 y RULE_TR_3.
¿Son precisos los algoritmos?	A-3, #7	6.3.1g	✔	No hay algoritmos propuestos.

Figura 50. Resultado de la lista de comprobación del proceso de requisitos

Checklist del documento Especificación de Requisitos Software (SRD)			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Asignación de los requisitos de sistema	11.9a		
¿Está descrita la asignación de requisitos de sistema al software?		✔	Todos los requisitos software identificados en la especificación de sistema corresponden a este elemento de configuración.
¿Están descritos los requisitos safety de sistema asignados al software?		✔	Los requisitos de safety están incluídos en el apartado 4.8 del documento.
¿Están descritas las condiciones potenciales de fallo?		✔	La contibución del software a las condiciones de fallo del sistema está descrita en el documento PSAC (0149900000000PL01)
Requisitos funcionales y operacionales	11.9b		
¿Están descritos los requisitos software funcionales y operacionales para cada modo de operación?		✔	Los requisitos funcionales están incluídos en el apartado 4.2, y los requisitos operacionales en el apartado 4.3
Requisitos de Precisión	11.9c		
¿Están definidos los requisitos de precisión y consistencia?		✔	La precisión está fijada por los interfaces externos.
Requisitos de Temporización	11.9d		
¿Están definidos los requisitos de temporización del software?		✔	La temporización de las tramas de datos está definida en el requisito SR1_INTE_0290. Existen otros requisitos de tiempos en el
Tamaño de memoria	11.9e		
¿Se han descrito las limitaciones del tamaño de memoria?		✔	Las limitaciones del tamaño de memoria vienen establecidas por el HW.
Interfaces hardware y software	11.9f		
¿Se han descrito los interfaces hardware y software?		✔	Al existir un único elemento de configuración, sólo se han definido los interfaces externos, en el apartado 3.1 del documento
¿Se han descrito los protocolos para los interfaces?		✔	Los protocolos con los interfaces externos están descritos en el apartado 3.1 del documento.
Monitorización y detección de fallos	11.9g		
¿Se han descrito los mecanismos de monitorización?		✔	Los mecanismos de monitorización están descritos en los apartados 3.3 y 4.16
¿Se han descrito los mecanismos de detección de fallos?		✔	Los mecanismos de detección de fallos están descritos en los apartados 3.3 y 4.16
Requisitos de particionamiento	11.9h		
¿Se han descrito las particiones del software?		✔	No existen particiones software
¿Se han descrito los requisitos control entre las distintas particiones?		✔	No existen particiones software

Figura 51. Resultado de la lista de comprobación para el SRD

Especificación de Requisitos Software (SRD)										
#	**Rev.**	**Edición**	**Regla**	**Req.**	**Pág.**	**Comentario de Verificación**	**Respuesta**	**Cerrado en Versión**	**Estado**	**Acciones**
52	JEB	A/0	RULE_TX_3	SR1_FUNC_0010	4-2	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado.	El uso del artículo en 'una disminución o alteración de las capacidades actuales' es aplicable a 'cualquier disminución o alteración de las capacidades actuales', por lo que el requisito es preciso.	N/A	Cerrada	
53	JEB	A/0	RULE_TX_3	SR1_FUNC_0020	4-2	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Se recomienda cambiar la frase 'un control redundante' por 'el control redundante' para mejorar la precisión/consistencia del requisito.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
54	JEB	A/0	RULE_TX_3	SR1_FUNC_0120	4-3	Acabar una lista con "etc" o "..." está desconsejado. No se ve clara la referencia o el significado de 'tareas normales' en el requisito.	La lista ya no acaba con "etc."	A/1	Cerrada	
55	JEB	A/0	RULE_TX_3	SR1_FUNC_0170	4-4	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Está recomendado el cambio de la frase 'integrará un conjunto de otras cuatro presentaciones' por 'el conjunto de otras cuatro presentaciones'.	El uso de 'un' en la frase 'un conjunto' se refiere a un conjunto determinado y definido dentro del mismo requisito.	N/A	Cerrada	
56	JEB	A/0	RULE_TX_3	SR1_FUNC_0180	4-4	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Está recomendado cambiar 'una lista de estado operativo' por 'la lista con el estado operativo', identificando así de forma única esta lista, que está descrita en la explicación que viene a continuación.	Requisito corregido	A/1	Cerrada	
57	JEB	A/0	RULE_TX_3	SR1_FUNC_0190	4-4	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Está recomendado cambiar 'una lista de las versiones del software' por 'la lista con las versiones del software', ya que es una lista perfectamente definida en la explicación.	Requisito eliminado	A/1	Cerrada	
58	JEB	A/0	RULE_TX_3	SR1_FUNC_0210	4-5	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Está recomendado el cambio de la expresión 'una página de presentación' por 'cualquier página de presentación'.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
59	JEB	A/0	RULE_TX_3	SR1_FUNC_0230	4-5	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Está recomendado cambiar 'un' por 'el', de tal forma que se especifica que todos los indicadores están perfectamente definidos.	Los contenidos de la Presentación Primaria de Vuelo (PFD) están definidos en los requisitos SR1_FUNC_0240, SR1_FUNC_0250 y SR1_FUNC_0260.	N/A	Cerrada	
60	JEB	A/0	RULE_TX_3	SR1_FUNC_0240	4-5	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "que incluye aviso del modo VNAV" por "que incluirá aviso del modo VNAV".	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
61	JEB	A/0	RULE_TX_3	SR1_FUNC_0240	4-5	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	

Figura 52. Comentarios registrados para el SRD (I)

Especificación de Requisitos Software (SRD)										
#	Rev.	Edición	Regla	Req.	Pág.	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
62	JEB	A/0	RULE_TX_3	SR1_FUNC_0250	4-5	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "que incluye la escala de cabeceo y alabeo del avión" por "que incluirá la escala del cabeceo y alabeo del avión".	El horizonte artificial siempre contiene la escala de cabeceo y alabeo del avión, por lo que esta frase es sólo aclaratoria para aumentar la precisión en la definición del requisito.	N/A	Cerrada	
63	JEB	A/0	RULE_TX_3	SR1_FUNC_0250	4-5	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
64	JEB	A/0	RULE_TX_3	SR1_FUNC_0260	4-6	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "Normalmente es de color verde, aunque cuando se piede la señal de la fuente de navegación se pone rojo, desapareciendo del HSI el Bearing Point No. 1." por "Será de color verde, excepto cuando se pierda la señal de la fuente de navegación, que se pondrá de color rojo, desapareciendo del HSI el Bearing Pointer No. 1." o una frase similar.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
65	JEB	A/0	RULE_TX_3	SR1_FUNC_0260	4-6	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
66	JEB	A/0	RULE_TX_3	SR1_FUNC_0280	4-7	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "Estos formatos son los siguientes" por "Estos formatos serán los siguientes".	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
67	JEB	A/0	RULE_TX_3	SR1_FUNC_0290	4-7	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado	El uso de 'un' en la frase 'un área dedicada' no reduce la precisión del requisito, ya que este área dedicada para la presentación de avisos está perfectamente definida.	A/1	Cerrada	
68	JEB	A/0	RULE_TX_3	SR1_FUNC_0300	4-7	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Se recomienda cambiar la frase "Cuando se produzca un nuevo aviso" por "Cuando se produzca cualquier nuevo aviso".	Requisito eliminado	A/1	Cerrada	
69	JEB	A/0	RULE_TX_3	SR1_FUNC_0310	4-8	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "esta información no está disponible en la SFD HSI" por "esta información no estará disponible en la SFD HSI" tantas veces como aparezca.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
70	JEB	A/0	RULE_TX_3	SR1_FUNC_0310	4-8	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase: "Normalmente es de color verde, aunque cuando se piede la señal de la fuente de navegación se pone rojo, desapareciendo del HSI el Bearing Point No. 1." por "Será de color verde, excepto cuando se pierda la señal de la fuente de navegación, que se pondrá de color rojo, desapareciendo del HSI el Bearing Pointer No. 1." o una frase similar.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
71	JEB	A/0	RULE_TX_3	SR1_FUNC_0310	4-8	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	

Figura 53. Comentarios registrados para el SRD (II)

Especificación de Requisitos Software (SRD)										
#	Rev.	Edición	Regla	Req.	Pág.	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
72	JEB	A/0	RULE_TX_3	SR1_FUNC_0320	4-8	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Se recomienda cambiar la frase "proporcionará una presentación ampliada" por "proporcionará la presentación ampliada"	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
73	JEB	A/0	RULE_TX_3	SR1_FUNC_0320	4-8	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
74	JEB	A/0	RULE_TX_3	SR1_FUNC_0320	4-8	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "se pueden presentar hasta dos indicaciones de ese tipo" por "se podrán presentar hasta dos indicaciones de ese tipo"	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
75	JEB	A/0	RULE_TX_3	SR1_FUNC_0330	4-9	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Se recomienda cambiar la frase "proporcionará una ampliación de un sector del HSI" por "proporcionará la ampliación de un sector del HSI".	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
76	JEB	A/0	RULE_TX_3	SR1_FUNC_0330	4-9	Acabar una lista con "etc." o "…" está desaconsejado.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
77	JEB	A/0	RULE_TX_3	SR1_FUNC_0330	4-9	El requisito deberá estar escrito en un futuro imperativo. Se recomienda cambiar la frase "las marcas serán verdes normalmente, salvo cuando el radar está conectado…" por "las marcas serán verdes, salvo cuando el radar está conectado…"	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
78	JEB	A/0	RULE_TX_3	SR1_FUNC_0330	4-9	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
79	JEB	A/0	RULE_TX_3	SR1_FUNC_0340	4-10	El uso del artículo indefinido "un/a" y "unos/as" no es recomendado. Se recomienda cambiar la frase "proporcionará una ampliación de un sector del HSI" por "proporcionará la ampliación de un sector del HSI".	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	
80	JEB	A/0	RULE_TX_3	SR1_FUNC_0340	4-10	La declaración de una lista deberá encontrarse numerada.	Requisito cambiado de acuerdo a la recomendación.	A/1	Cerrada	

Figura 54. Comentarios registrados para el SRD (III)

El documento de requisitos software (SRD) es uno de los más importantes y que más cosas hay que tener en cuenta a la hora de realizar la verificación, ya que los requisitos definidos en él van a servir de base para realizar el diseño y especificar las pruebas. Por ello, el documento tiene muchos comentarios asociados, y en esta memoria sólo se presentan algunos de ellos.

Como se puede ver, en este caso hay comentarios que han conllevado cambios en el documento y otros que, tras una aclaración por parte del autor o responsable del proceso, no han derivado en modificaciones.

Para la revisión, se tomó en cuenta tanto la normativa aplicable (DO-178B, Ref. [4]) como el estándar de requisitos definido en el proyecto. Este estándar ya fue revisado para cumplimiento con la normativa en el proceso de planificación.

4.2.2.4. Proceso de Diseño

Checklist del proceso de Diseño				
Objetivo	**Objetivo de la Tabla**	**Referencia DO-178B**	**¿Cumple?**	**Comentarios**
¿Los requisitos software de bajo nivel cumplen con los requisitos software de alto nivel?	A-4, #1	6.3.2a	✔	Los requisitos software de alto nivel están trazados con los requisitos software de bajo nivel. La trazabilidad ha sido revisad
¿Es compatible la arquitectura software con el hardware?	A-4, #10	6.3.3c	✔	Se ha tenido en cuenta el hardware para la arquitectura software y es compatible.
¿Es verificable la arquitectura software?	A-4, #11	6.3.3d	✔	Es verificable.
¿Cumple la arquitectura software con los estándares?	A-4, #12	6.3.3e	✔	La arquitectua software se ha revisado con respecto a los estándares y cumple.
¿Está confirmada la integridad de las particiones software?	A-4, #13	6.3.3f	✔	No existen particiones software.
¿Los requisitos software de bajo nivel son precisos y consistentes?	A-4, #2	6.3.2b	✔	Se han revisado con respecto al estándar para su precisión y consistencia.
¿Los requisitos software de bajo nivel son compatibles con el hardware?	A-4, #3	6.3.2c	✔	Se ha tenido en cuenta el hardware para la arquitectura y es compatible.
¿Los requisitos software de bajo nivel son verificables?	A-4, #4	6.3.2d	✔	Son verificables.
¿Los requisitos software de bajo nivel cumplen con los estándares?	A-4, #5	6.3.2e	✔	Los requisitos software de bajo nivel cumplen con los estándares.
¿Los requisitos software de bajo nivel están trazados a los requisitos software de alto nivel?	A-4, #6	6.3.2f	✔	Las matrices de trazabilidad están incluidas en los apéndices A y B, y contenida en Requisite Pro.
¿Son precisos los algoritmos?	A-4, #7	6.3.2g	✔	Son precisos.
¿Es compatible la arquitectura software con los requisitos software de alto nivel?	A-4, #8	6.3.3a	✔	Es compatible.
¿Es consistente la arquitectura software?	A-4, #9	6.3.2b	✔	Es consistente.

Figura 55. Resultado de la lista de comprobación del proceso de diseño

Checklist del documento Documento de Diseño Software (SDD)			
Objetivo	**Referencia DO-178B**	**¿Cumple?**	**Comentarios Verificación**
Cumplimiento con los requisitos de alto nivel	11.10a		
¿Se ha descrito cómo el software satisface los requisitos de alto nivel?		✔	La descripción está incluida en el apartado 4.
Descripción de la arquitectura software	11.10b		
¿Se ha descrito la arquitectura software?		✔	La arquitectura software está definida en el apartado 5 del documento.
Descripción de las entradas y salidas	11.10c		
¿Se han descrito las entradas y salidas a través de la arquitectura software? Esto incluye, por ejemplo, los diccionarios de datos.		✔	Las entradas y las salidas están descritas en el apartado 6 del documento.
Flujo de control y de datos	11.10d		
¿Se ha descrito el flujo de control y de datos del diseño?		✔	El flujo de control y flujo de datos están descritos en el aparado 7.
Limitaciones en los recursos	11.10e		
¿Se han descrito las limitaciones de los recursos? Por ejemplo la temporización y la memoria.		✔	La descripción del mapa de memoria necesario está descrito en el apartado 8.
Procesos de planificación de tareas	11.10f		
¿Se han descrito los procesos de planificación de tareas?		✔	La planificación temporal de la ejecución está incluida en el apartado 9.
Métodos de diseño	11.10g		
¿Se han descrito los métodos de diseño y detalles para su implementación?		✔	Se hace una referencia a los métodos de diseño utilizados en el apartado 10 del documento.
Métodos de particionamiento	11.10h		
¿Se han definido los mecanismos de protección de las particiones?		✔	La aplicación no tiene particiones: existe sólo una.
Descripción de los componentes software	11.10i		
¿Se han descrito los componentes software? Si son previamente desarrollados, es necesario especificar información acerca de su origen.		✔	La descripción de los componentes software se incluye en el apartado 12.
Requisitos derivados	11.10j		
¿Se han definido y justificado los requisitos derivados del proceso de diseño software?		✔	No existen requisitos derivados, tal y como se establece en el apartado 13 del documento.
Tratamiento del código desactivado	11.10k		
Si el sistema contiene código desactivado, ¿existe una descripción del modo en que se asegura que este código no va a ser ejecutado en el sistema?		✔	No existe código desactivado, tal y como se establece en el apartado 14 del documento.
Decisiones de diseño	11.10l		
¿Existe una justificación de las decisiones de software relacionadas con los requisitos safety?		✔	Las decisiones de diseño se explican durante todo el documento, y se incluye un resumen en el apartado 15

Figura 56. Resultado de la lista de comprobación para el SDD

Documento de Diseño Software (SDD)										
#	**Rev.**	**Edición**	**Regla**	**Req.**	**Pág.**	**Comentario de Verificación**	**Respuesta**	**Cerrado en Versión**	**Estado**	**Acciones**
178	JEB	B/0	RULE_DB_1	Todos	-	Los requisitos no tienen todos los atributos obligatorios definidos.	Se han definido los atributos obligatorios donde no existían y se ha tenido en cuenta para la creación de los nuevos requisitos	D/0	Cerrada	
179	JEB	B/0	RULE_DB_3	Todos	-	No está definido el método de verificación.	Corregido en la nueva edición del documento.	D/0	Cerrada	
180	JEB	B/0	RULE_TX_3	Todos	-	Es necesario revisar todos los requisitos con respecto a esta regla.	Corregido en la nueva edición del documento.	D/0	Cerrada	
181	JEB	B/0	RULE_DD_03	-	-	El flujo de control no está definido.	Incluído en la nueva edición del documento.	D/0	Cerrada	
182	JEB	B/0	RULE_DD_04	-	-	El flujo de datos no está definido.	Incluído en la nueva edición del documento.	D/0	Cerrada	
183	JEB	B/0	RULE_LL_01	Todos	-	Los requisitos de bajo nivel no están trazados.	Incluído en la nueva edición del documento.	D/0	Cerrada	
184	JEB	B/0	RULE_LL_05	Todos	-	Los requisitos software de bajo nivel no están asignados a componentes software.	Incluído en la nueva edición del documento.	D/0	Cerrada	
185	JEB	B/0	RULE_LL_06	Todos	-	Los requisitos software de bajo nivel no están asignados a componentes software.	Incluído en la nueva edición del documento.	D/0	Cerrada	
186	JEB	B/0	RULE_LL_07	-	-	No hay componentes software definidos expresamente.	Incluído en la nueva edición del documento.	D/0	Cerrada	
187	JEB	B/0	RULE_TX_1	Todos	-	Es necesario revisar todos los requisitos para incluir los acrónimos necesarios.	Incluído en la nueva edición del documento.	D/0	Cerrada	
188	JEB	D/0*	-	-	8-1	El mapa de memoria no es correcto	Se ha actualizado el mapa de memoria con los datos correctos.	D/0	Cerrada	

Figura 57. Comentarios registrados para el SDD

En el proceso de diseño se verifica entre otras cosas la arquitectura software y los requisitos software de bajo nivel. Esto se revisa de acuerdo a la normativa aplicable como al Estándar de Diseño definido para el proyecto. Las primeras versiones del documento carecían de gran parte de la información básica requerida, tal como el flujo de control, el flujo de datos y la trazabilidad

de los requisitos de bajo nivel. Esta información se incluyó en la última versión del documento (Edición/Revisión D/0).

4.2.2.5. Proceso de Implementación

En el proceso de implementación se revisó el código fuente y su relación con la documentación de diseño.

Checklist del proceso de Implementación				
Objetivo	**Objetivo de la Tabla**	**Referencia DO-178B**	**¿Cumple?**	**Comentarios**
¿Está trazado el código fuente con los requisitos software de bajo nivel?	A-4, #5	6.3.4e	✔	El código fuente está trazado a los requisitos software de bajo nivel en la herramienta Requisite Pro, a través de la introdu
¿Es preciso y consistente el código fuente?	A-4, #6	6.3.4f	✔	De acuerdo con la revisión del código fuente según el estándar de codificación, es preciso y consistente.
¿Está de acuerdo el código fuente con los requisitos software de bajo nivel?	A-5, #1	6.3.4a	✔	El código fuente cumple con los requisitos de bajo nivel.
¿Está de acuerdo el código fuente con la arquitectura software?	A-5, #2	6.3.4b	✔	El código fuente está desarrollado de acuerdo a la arquitectura software.
¿Es verificable el código fuente?	A-5, #3	6.3.4c	✔	Sí. Se ha realizado un análisis estático y dinámico del mismo.
¿Cumple el código fuente con los estándares?	A-5, #4	6.3.4d	✔	La revisión con respecto al estándar indica que el código fuente cumple con el estándar de codificación.
¿Es completa y correcta la salida del proceso de integración software?	A-5, #7	6.3.5	✔	El código objeto se ha probado dentro de la plataforma hardware con resultado satisfactorio.

Figura 58. Resultado de la lista de comprobación del proceso de implementación

Debido a que el código fuente es muy extenso y, por tanto, susceptible de muchas no conformidades, se encontraron muchos comentarios. Por ello, las siguientes tablas muestran *sólo* un extracto de todas las no-conformidades detectadas en el código fuente.

SW HERCULES SAF C-130 DU										
#	Rev.	Versión	Regla	Fichero	Línea de Código	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
189	JEB	1.0	REGLA_C_28	acceso_hw_tarjeta.c	114	Nombre de la macro no válido. Debe estar en mayúsculas	Revisado y corregido.	1.1	Cerrada	
190	JEB	1.0	REGLA_GEN_	acceso_hw_tarjeta.c	2624	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
191	JEB	1.0	REGLA_GEN_	acceso_hw_tarjeta.c	2008	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
192	JEB	1.0	REGLA_GEN_08	acceso_hw_tarjeta.c	733	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
193	JEB	1.0	REGLA_GEN_08	acceso_hw_tarjeta.c	648	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
194	JEB	1.0	REGLA_GEN_08	ahw_ip_cronometro.c	178	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
195	JEB	1.0	REGLA_GEN_08	ahw_ip_pseudo_asm.h	1034	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
196	JEB	1.0	REGLA_C_28	ahw_ip_pseudo_asm.h	212	Nombre de la macro no válido. Debe estar en mayúsculas	Revisado y corregido	1.1	Cerrada	
197	JEB	1.0	REGLA_GEN_08	ahw_ip_pseudo_asm.h	163	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
198	JEB	1.0	REGLA_GEN_08	ahw_ip_pseudo_asm.h	498	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	

Figura 59. Comentarios registrados para el Código Fuente (I)

SW HERCULES SAF C-130 DU										
#	Rev.	Versión	Regla	Fichero	Línea de Código	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
199	JEB	1.0	REGLA_GEN_08	ahw_ip_pseudo_asm.h	797	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
200	JEB	1.0	REGLA_GEN_08	ahw_ip_pseudo_asm.h	947	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
201	JEB	1.0	REGLA_GEN_08	bite.c	1737	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
202	JEB	1.0	REGLA_GEN_08	bite.c	1066	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
203	JEB	1.0	REGLA_GEN_08	control_429_info_general.c	2508	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
204	JEB	1.0	REGLA_GEN_08	modo_comandos_422.c	504	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
205	JEB	1.0	REGLA_GEN_08	Pantalla_PFD_Zona_HSI.c	1156	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
206	JEB	1.0	REGLA_GEN_08	respuestas_DU_a_MC.c	1797	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
207	JEB	1.0	REGLA_GEN_08	tabla_mensaje.c	6427	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
208	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	19390	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	

Figura 60. Comentarios registrados para el Código Fuente (II)

SW HERCULES SAF C-130 DU										
#	Rev.	Versión	Regla	Fichero	Línea de Código	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
239	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	20116	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
240	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	20037	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
241	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	18943	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
242	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	18916	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
243	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	19262	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
244	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	19295	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
245	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	19055	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
246	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	19010	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
247	JEB	1.0	REGLA_GEN_08	valores_matematicas.c	19109	Debe haber un espacio después de la coma	Revisado y corregido	1.1	Cerrada	
248	JEB	1.1	REGLA_C_05	ahw_ip_cronometro.h	45	El nombre de los Typedef deben empezar por T_ y estar en mayúsculas	Revisado y corregido	1.6	Cerrada	

Figura 61. Comentarios registrados para el Código Fuente (III)

SW HERCULES SAF C-130 DU										
#	Rev.	Versión	Regla	Fichero	Línea de Código	Comentario de Verificación	Respuesta	Cerrado en Versión	Estado	Acciones
249	JEB	1.1	REGLA_C_18	ahw_ip_rs232.h	311	Función sin uso	Función eliminada	1.6	Cerrada	
250	JEB	1.1	REGLA_GEN_08	respuestas_DU_a_MC.c	1320	Debe haber un espacio después de la coma	Revisado y corregido	1.6	Cerrada	
251	JEB	1.6	REGLA_C_11	bite.c	0	Faltan las condiciones else.	Revisado y corregido	1.7	Cerrada	
252	JEB	1.6	REGLA_GEN_08	respuestas_DU_a_MC.c	12	Variable local sin uso	Variable eliminada	1.7	Cerrada	
253	JEB	1.8	REGLA_C_18		116	Función sin uso	Función eliminada	1.9	Cerrada	
254	JEB	1.12	REGLA_GEN_08	bite.c	1300	Debe haber un espacio después de la coma	Revisado y corregido	1.13	Cerrada	

Figura 62. Comentarios registrados para el Código Fuente (IV)

El análisis del código fuente con respecto a las reglas de codificación se realiza principalmente de forma estática (sin necesidad de ejecutar), y en muchos casos con ayuda de herramientas de análisis del código. A pesar del uso de herramientas, existen otras reglas que necesariamente tienen que ser comprobadas manualmente.

1.4.3.1. Proceso de Integración

Checklist del proceso de Integración				
Objetivo	Objetivo de la Tabla	Referencia DO-178B	¿Cumple?	Comentarios
¿Cumple el código objeto ejecutable con los requisitos software de alto nivel?	A-6, #1	6.4.2.1, 6.4.3	✔	Los resultados de las pruebas se encuentran en el documento Informe de los Resultados de las Pruebas de Calificación Software (
¿Es robusto el código objeto ejecutable con los requisitos software de alto nivel?	A-6, #2	6.4.2.2, 6.4.3	✔	Los resultados de las pruebas de robustez están incluidas en el documento Informe de los Resultados de las Pruebas de Calificac
¿Cumple el código objeto ejecutable con los requisitos software de bajo nivel?	A-6, #3	6.4.2.1, 6.4.3	✔	Los resultados de las pruebas se encuentran en el documento Informe de Resultados de las Pruebas Unitarias Software (SUTR)
¿Es robusto el código objeto ejecutable con los requisitos software de bajo nivel?	A-6, #4	6.4.2.2, 6.4.3	✔	Los resultados de las pruebas de robustez se encuentran en el documento Informe de Resultados de las Pruebas Unitarias Software
¿Es compatible el código objeto ejecutable con el hardware?	A-6, #5	6.4.3a	✔	Las pruebas se han ejecutado en la plataforma hardware definitiva.

Figura 63. Resultado de la lista de comprobación del proceso de Integración

En el proceso de integración se verifica el cumplimiento del código objeto ejecutable con los requisitos software de alto y bajo nivel. Se trata por tanto de un análisis dinámico realizado a través de pruebas, que están reflejadas en los documentos correspondientes.

1.4.3.2. Proceso de Verificación

Checklist del proceso de Verificación				
Objetivo	**Objetivo de la Tabla**	**Referencia DO-178B**	**¿Cumple?**	**Comentarios**
¿Son correctos los procedimientos de pruebas?	A-7, #1	6.3.6b	✔	Descrito dentro de las pruebas definidas en el Plan de Pruebas de Calificación Software (STP) - 01499000000000TP01 y Plan de Pru
¿Son correctos los resultados de las pruebas y están explicadas las discrepancias?	A-7, #2	6.3.6c	✔	Los resultados de las pruebas están contenidos en los documentos Informe de los Resultados de las Pruebas de Calificación Soft
¿Se ha conseguido la cobertura completa de las pruebas para los requisitos software de alto nivel?	A-7, #3	6.4.4.1	✔	Todas las pruebas han sido ejecutadas con resultado satisfactorio, y la trazabilidad está contenida en Requisite Pro.
¿Se ha conseguido la cobertura completa de las pruebas para los requisitos software de bajo nivel?	A-7, #4	6.4.4.1	✔	Todas las pruebas han sido ejecutadas con resultado satisfactorio, y la trazabilidad está contenida en Requisite Pro.
¿Se ha conseguido la cobertura estructural completa de decisiones?	A-7, #6	6.4.4.2a, 6.4.4.	✔	Ver Informe de Resultados de las Pruebas Unitarias Software (SUTR)
¿Se ha conseguido la cobertura estructural completa de decisiones?	A-7, #7	6.4.4.2a, 6.4.4.	✔	Ver Informe de Resultados de las Pruebas Unitarias Software (SUTR)
¿Se ha conseguido la cobertura estructural completa (flujo de control y flujo de datos)?	A-7, #8	6.4.4.2c	✔	Ver Informe de Resultados de las Pruebas Unitarias Software (SUTR)

Figura 64. Resultado de la lista de comprobación del proceso de Verificación

La verificación del proceso de verificación comprobó que todas las actividades se llevaron a cabo correctamente. En este caso, se indica que las pruebas están correctamente definidas y los resultados son correctos. No existieron discrepancias en el proceso.

CAPÍTULO 5: CONCLUSIONES

En este capítulo se presentan las conclusiones después de la realización de este trabajo, proponiendo ampliaciones futuras y terminando con un resumen del proyecto.

5.1. CONCLUSIONES ACERCA DE LA IMPLEMENTACIÓN ACTUAL

La falta de formación en los procesos de las normativas aplicables en los sistemas con requisitos safety hace que los costes y las dificultades para el cumplimiento de los requisitos aumenten. Esta herramienta clarifica estos procesos y proporciona un entorno de trabajo colaborativo que facilita la aplicación de las normativas de verificación software.

En AVeMaCS se ha plasmado el ciclo de vida en "V", y se han definido una serie de documentos para cada uno de las etapas. Gracias a esto, las personas involucradas en los proyectos pueden conocer el ciclo de vida definido para el proyecto con un vistazo en la herramienta, con lo que el proceso de formación disminuye.

Para cada uno de los elementos se puede ver las no conformidades detectadas, lo que permite la corrección temprana de las mismas por parte del responsable correspondiente.

De esta forma, la aplicación de la normativa se puede realizar desde el inicio y los costes de adaptación se reducen considerablemente.

Debido a la tecnología utilizada no se necesitan desembolsos adicionales de dinero, puesto que todos los participantes tendrán el entorno necesario para poder hacer uso de la herramienta.

Repasando los objetivos propuestos al principio de esta memoria, se llegan a las conclusiones expuestas en los siguientes subapartados.

5.1.1. Cubrir las actividades de gestión de la verificación

La herramienta cubre todos los procesos de verificación de acuerdo a los distintos niveles aplicables y las necesidades de cada proyecto. Se definen por defecto las normativas DO-178B y ED-109 y se han creado las listas de comprobación y los documentos que se utilizan normalmente en los proyectos de este tipo.

5.1.2. Desarrollo de un entorno colaborativo

El uso de la herramienta AVeMaCS se realiza mediante un entorno web. De esta forma, lo único que necesitan los usuarios es acceso al servidor donde se encuentre la aplicación y un navegador web. Esta forma de acceso pretende facilitar al máximo la adopción de la herramienta como parte del proceso del ciclo de vida de los proyectos.

5.1.3. Facilitar la integración con el resto de procesos del ciclo de vida

Bajo el objetivo de la integración de todas las tareas de verificación y disponibilidad de la información en un entorno centralizado, se pueden proponer ampliaciones para la facilitar integración con otros procesos del ciclo de vida de desarrollo software.

Esta herramienta se trata principalmente de una herramienta de Gestión de la verificación, de tal forma que ofrece un lugar centralizado de todas las actividades de verificación que se llevan a cabo para distintos proyectos.

Una arquitectura propuesta para ampliaciones futuras se puede ver en la Figura 65.

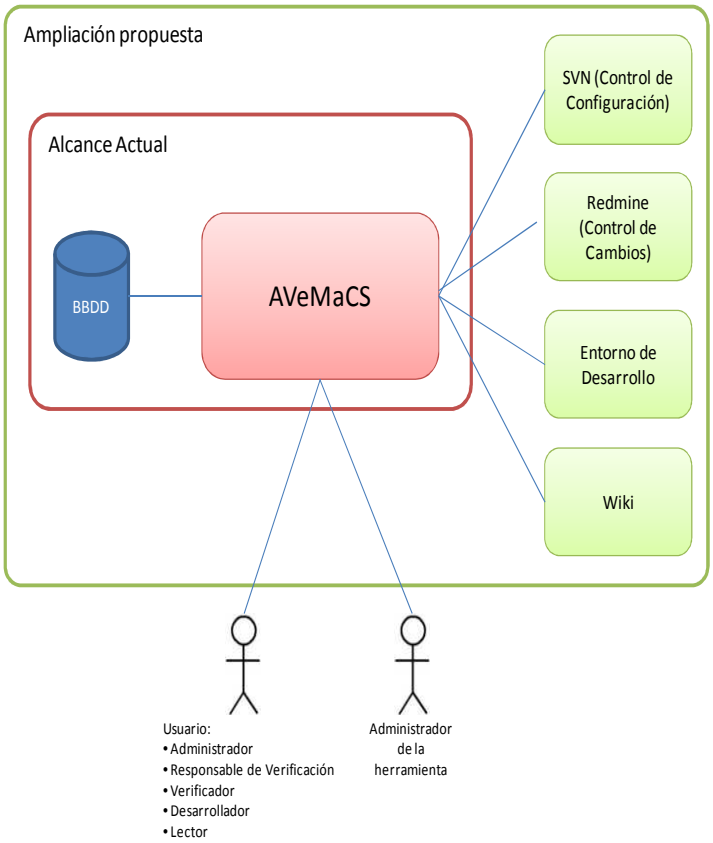


Figura 65. Posible ampliación de AVeMaCS

La descripción de la ampliación propuesta se describe en los siguientes puntos:

- Integración con el control de configuración

Una de las posibles ampliaciones de la herramienta, sería conectarla con el repositorio de control de configuración donde se almacenen todos los elementos de configuración. De esta forma, sería posible enlazar directamente los distintos comentarios, listas de comprobación y referencias a los documentos almacenados bajo control de configuración, donde se encontrarían las distintas versiones de los mismos. Una posibilidad sería utilizar la herramienta SubVersion (SVN).

Subversion es una herramienta de control de versiones, bajo una licencia de tipo Apache/BSD. Subversion puede acceder al repositorio a través de redes, lo que permite ser usado en un entorno colaborativo.

- Integración con el control de cambios

Sería posible conectar AVeMaCS con las herramientas de control de cambios, de tal forma que los comentarios registrados para los elementos de configuración que requieran un cambio se pudieran trazar directamente a las incidencias o PCIs registradas. Como herramienta de control de cambios, se puede utilizar Redmine.

Redmine es una herramienta para la gestión de proyectos, que incluye un sistema de seguimiento de incidencias entre otras funcionalidades. Se trata de software libre y de código abierto, disponible bajo Licencia Pública General de GNU v2.

- Integración con las herramientas de desarrollo

En el caso de que la herramienta de desarrollo permita alguna actividad de verificación, se podría implementar una integración con la misma. Sería el caso, por ejemplo, de un entorno de desarrollo basado en Eclipse. Para ciertos niveles de certificación es necesario realizar un análisis estático del código, basado en las reglas de codificación especificadas en el estándar. En el caso de que exista un plugin de Eclipse u otro tipo de herramienta que analice el código de acuerdo a esas reglas de codificación, habría dos opciones para registrar las no conformidades detectadas:

- Exportar el resultado del análisis estático automático a AVeMaCS
- Crear un enlace en AVeMaCS a los informes de resultados de las herramientas.

De esta forma se podría generar este informe automáticamente y no sería necesario registrar estas no conformidades manualmente.

- Integración con wikis

En ocasiones, las no conformidades encontradas son aclaradas por el equipo autor del elemento de configuración, de tal forma que la no conformidad no requiere ninguna modificación. Estas "conversaciones", que pueden requerir un nivel de detalle mayor al que se incluye en AVeMaCS, podrían incluirse en una wiki integrada.

Sería posible integrar una wiki en AVeMaCS, de tal forma que quede registro del proceso de revisión y acuerdo por parte de los verificadores y los autores de los distintos elementos de configuración.

5.1.4. Posibilidad de ampliaciones futuras

Aunque se han definido unas normativas básicas, la herramienta es personalizable, de tal forma que las listas de comprobación (tanto de los procesos como de los documentos) son personalizables, así como los documentos que pertenecen a cada proyecto.

De esta forma, es posible introducir fácilmente normativas nuevas, definiendo los objetivos, documentos y listas de comprobación particulares para cada caso.

Desde el punto de vista de una ampliación del alcance, con un desarrollo adicional, se podría introducir el uso de métricas. Las métricas deben formar parte del sistema de Calidad de los proyectos. Las métricas dan una idea del estado y evolución de un proyecto, así como una medida de la Calidad de los mismos. Además, sirven como base para cubrir manuales de buenas prácticas, por ejemplo en niveles altos de madurez CMMi.

Gracias a la utilización de una base de datos, sería prácticamente inmediato extraer algunas métricas, como por ejemplo número de no conformidades por proyecto y número de no conformidades totales.

5.2. RESUMEN DEL PROYECTO

Durante el desarrollo del proyecto, se han tenido en cuenta las necesidades en el ciclo de vida de desarrollo software de un proyecto con implicaciones en la seguridad. Se han analizado estas necesidades y se ha implementado una herramienta de gestión de la verificación. Esta herramienta debía tener la posibilidad de personalización suficiente para ajustarse a las necesidades de distintos proyectos, dependiendo de varios factores, como son el nivel de aseguramiento requerido, de la experiencia previa con el correspondiente cliente o de las distintas metodologías que se puedan utilizar en distintos departamentos o mercados.

Se han cubierto los objetivos que se plantearon al inicio del proyecto, y se han propuesto posibles mejoras y ampliaciones futuras que pueden servir como base para cubrir parte de nuevos procesos dentro del ciclo de vida.

"Página dejada en blanco intencionadamente"

CAPÍTULO 6: BIBLIOGRAFÍA

Este capítulo contiene una lista de la bibliografía utilizada para el desarrollo de este proyecto.

[1] IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems. IEC, Abril 2010.

[2] "Out of Control: Why Control Systems Go Wrong and How to Prevent Failure". Editor: Health and Safety Executive (HSE). Segunda Edición, 2003. ISBN 978-0-7176-2192-7

[3] "No Silver Bullet – Essence and Accidents of Software Engineering". Autor: Brooks, F.P., Jr. Artículo de la revista "Computer" (IEEE). Abril 1987.

[4] RTCA DO-178B / EUROCAE ED-12B: Software Considerations in Airborne Systems and Equipment Certification. RTCA, Inc. 1992

[5] RTCA DO-278 / EUROCAE ED-109: Guidelines for Communication, Navigation Surveillance, and Air Traffic Management (CNS/ATM) Systems Software Integrity Assurance. RTCA Inc, 03/05/2002.

[6] RTCA DO-254 / EUROCAE ED-80: Design Assurance Guidance for Airborne Electronic Hardware. RTCA Inc. 2000.

[7] ESARR6: Eurocontrol Safety Regulatory Requirement 6 – Software in ATM Functional Systems. Eurocontrol, 06/05/2010.

[8] ESARR4: Eurocontrol Safety Regulatory Requirement 4 – Risk Assessment and Mitigation in ATM. Eurocontrol, 05/04/2001.

[9] EUROCAE ED-153: Guidelines for ANS Software Safety Assurance. EUROCAE, Agosto 2009.

[10] RTCA DO-178C / EUROCAE ED-12C: Software Considerations in Airborne Systems and Equipment Certification. RTCA, Inc. 12/13/2011.

[11] RTCA DO-278A / EUROCAE ED-109A: Software Integrity Assurance Considerations for Communication, Navigation, Surveillance and Air Traffic Management (CNS/ATM) Systems. RTCA Inc, 12/13/2011.

[12] CENELEC 50128:2001: "Railway applications - Communications, signalling and processing systems". CENELEC, 2001.

[13] ISO 26262: Road Vehicles – Functional Safety. ISO, 11/11/2011.

[14] IEC 62304: Medical device software – Software life cycle processes. IEC, Mayo 2006.

[15] MIL-STD-498. Military-Standard-498. "Software development and documentation". Departamento de Defensa de Estados Unidos. Diciembre 1994.

[16] DOD-STD-2167A. Department of Defense Standard 2167A. "Defense system software development". Departamento de Defensa de Estados Unidos. Junio 1985.

[17] ISO/IEC 12207:2008 / IEEE Std 12207-2008. "Systems and software engineering – Software life cycle processes". ISO/IEC, 2008. IEEE, 2008.

[18] DEF STAN 00-55: "Requirements for Safety Related Software in Defence Systems". Ministerio de Defensa de Gran Bretaña. Agosto 1997.

[19] AC 20-115B: "Radio Technical Commission for Aeronautic, Inc. Document RTCA/DO-178B". FAA, Enero 1993.

[20] "Avionics Certification. A Complete Guide to DO-178 (Software), DO-254 (Hardware)". Autores: Vance Hilderman y Tony Baghai. Editor: Avionics Communications Inc. Primera Edición, 2008. ISBN 978-1-885544-25-4.

[21] "PHP Cookbook". Autores: David Sklar y Adam Tachtenberg. Editorial O'Reilly. Segunda Edición, Agosto 2006. ISBN-10: 0-596-10101-5, ISBN-13: 978-0-596-10101-5.

[22] "MySQL Developer's Library". Autor: Paul DuBois. Editorial Addison-Wesley. Cuarta Edición, Agosto 2008. ISBN-10: 0-672-32938-7, ISBN-13: 978-0-672-32938-8.

"Página dejada en blanco intencionadamente"

CAPÍTULO 7: PRESUPUESTO ESTIMADO

El presente capítulo contiene un presupuesto estimado para llevar a cabo el desarrollo de la herramienta que describe la memoria del proyecto AVeMaCS.

Para el desarrollo de esta herramienta, se han identificado las siguientes actividades:

- Especificación y definición de necesidades de la herramienta
- Desarrollo de la herramienta
- Documentación de la herramienta

A continuación se detalla cada una de las tareas:

- **Especificación y definición de necesidades de la herramienta.** Se definirán los objetivos a cumplir por la herramienta. Se realizará una primera especificación de la herramienta, que describa las necesidades para cumplimiento con los objetivos definidos. Esta tarea incluirá también la supervisión de los trabajos especificados en las otras dos tareas definidas, así como de la consecución o no de los objetivos definidos.
- **Desarrollo de la herramienta.** Desarrollo de la herramienta en PHP y MySQL en base a la especificación y objetivos definidos.
- **Documentación de la herramienta.** Documentación relativa al diseño, uso y propuestas de mejora de la herramienta.

Los recursos necesarios y tiempos que se han estimado para la realización de las tareas es la siguiente:

- Ingeniero Sénior: 2 meses. Coste: 8.579,08€
- Ingeniero Desarrollo: 10 meses. Coste: 26.943,90€
- 2 PCs uso dedicado al proyecto durante 10 meses al 100%: 233,33€

Tasa costes indirectos: 20% (7.151€)

Coste total: **42.908€.**

Este coste no incluye imprevistos como viajes o dietas.